

MAY 23 1981
MAY 2 1981

Item 835 H-15

NAS 1.60:1809

NASA Technical Paper 1809

ORIGINAL
COMPLETED

MEGA16 - Computer Program for Analysis and Extrapolation of Stress-Rupture Data

C. Robert Ensign

MAY 1981

NASA

(47)

NASA Technical Paper 1809

MEGA16 - Computer Program for Analysis and Extrapolation of Stress-Rupture Data

C. Robert Ensign
*Lewis Research Center
Cleveland, Ohio*



National Aeronautics
and Space Administration

**Scientific and Technical
Information Branch**

1981

Summary

The computerized form of the minimum commitment method of interpolating and extrapolating stress versus time to failure data is a program called MEGA16. This report describes MEGA16, giving examples of its many plots and tabular outputs for a typical set of data. The program assumes a specific model equation and then provides a family of predicted isothermals for any set of data with at least 12 stress-rupture results from three different temperatures spread over reasonable stress and time ranges. It is written in Fortran IV using IBM plotting subroutines, and it runs on an IBM 370 time sharing system.

Introduction

Fundamental to current design practice of ground power turbine equipment is an estimate of a maximum allowable stress at which a particular material can be expected to survive for at least 100 000 hours (11.6 yr) of exposure to a specified temperature. The search for a simple, reliable method of predicting such an estimated stress has been pursued for many years, by many individuals, using many scientific and empirical techniques (refs. 1 and 2). One of the more common approaches involves the use of a time-temperature parameter to extrapolate data from the standard stress-rupture test results. In such a method a model equation relating observed rupture life to temperature and stress is assumed, and its unknown terms are estimated using available test data. Characteristically, the data for a particular material are sparse, cover limited ranges of stress and temperature, or else are collected from a mixture of different heats, product forms, or other manufacturing and testing variables. In addition, the model itself introduces other assumptions and oversimplifications. Nonetheless, a time-temperature parameter is expected to furnish a reasonable estimate of maximum allowable stress at a given life or life at a given stress. This can be used in the design of machine components that are required to operate at high temperatures for extended periods.

The purpose of this paper is to present the computer program, MEGA16 (Manson Ensign Generalized Analysis, Version 16), which has been developed as an objective, fast, and reliable way to use a particular time-temperature method. The

method, often referred to as the "minimum commitment method," has been described previously (ref. 3). Results from its application to many sets of data using MEGA16 have been published (ref. 4); they compare favorably with results from other manual and computerized methods. A complete listing of MEGA16 and its subroutines is given in the appendix.

Program Description

General Features

Basically, MEGA16 is a computer program which fits, to one or many sets of input data, a model equation of the form

$$\log t + A\varnothing \log t + \varnothing = \mathcal{G}$$

where \varnothing and \mathcal{G} are temperature and stress functions, each with several coefficients to be determined. MEGA16 produces a series of plots that represent these functions and the stress rupture behavior of the material over the range of input data and to an extrapolated time to failure of up to 100 000 hours.

There are three sections to the main program (fig. 1): The first contains all the general housekeeping (dimensions, constants, etc.) and reading of data and variables associated with a given set of data. The second has the main loop which sets up and solves the equations for each value of an adjustable parameter called A , calculates the predicted values and statistics, prints the tabular results, and plots the curves for that A value. The third section produces other ancillary plots to show the merits of various A values used and the residuals of logarithm of time to failure versus other variables to indicate any trends that may have resulted with the model equation and the given data.

After looping through the various A values selected in the second section of the program (plus an extra A value read in the data) and doing the final plots in the third section, the program then goes on to the next set of data for another material and repeats the whole procedure. Thus a great number of data sets may be handled with one loading of the program.

Published stress-rupture data usually include fewer than 100 specimens for a material. The present form of MEGA16 can handle 200 data points per set; this

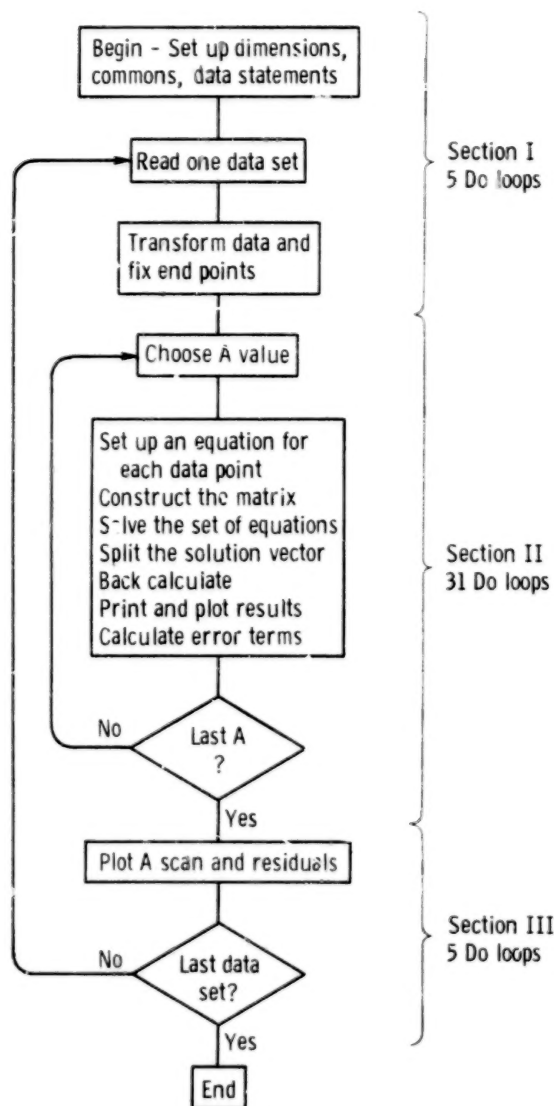


Figure 1. - Flowchart for MEGA16.

value could be increased, if necessary, by appropriate expansion of dimensions. The maximum size of the matrix to be inverted is 200 by 48. For this model equation the 48 dimension could be reduced considerably since there are not as many terms as in other model equations using this same framework. Since MEGA16 has been developed on a computer which has virtual memory, minimal effort was devoted to minimize storage requirements.

Input

Figure 2 is an example, with annotations, of the standard input data format required by MEGA16. The format is not efficient in terms of storage space required, but it has been very convenient for sorting, editing, and changing the constants of the control

cards and raw data. It also affords considerable room for expansion or addition of other information, such as creep behavior, tensile strength, or even comments about a particular data point. The first nine lines of every data set give the title, counters, constants, plotting values, and the number of data that follow. Each datum is then given on succeeding lines, with (from left to right) the values for temperature (in deg F), stress (in ksi), hours to failure, and number of times the point is to be weighted, along with other comments or information that will not be read by the program. The data are usually listed in the order of increasing temperature and decreasing stress. Also, all data with the same temperature are grouped together, while data with the same temperature and stress value are grouped in order of increasing life. Of course these formats could be changed to match a given set of data, but we have found it preferable to use a second program which transforms data from another format to this format. At the present, many data sets, for over 30 different materials, are stored on tape in this form, ready to run.

Calculations

The general equation used for MEGA16 (ref. 4) is

$$\log t + A \log t \cdot \Phi(T) + \Phi(T) = \mathcal{G}(\log S) \quad (1)$$

where

t	time to rupture, hr
A	a constant
$\Phi(T)$	a function for discrete temperature values
$\mathcal{G}(\log S)$	a stress function
S	σ/σ_0
σ	a given stress value
σ_0	a reference stress, near midrange

The constant A takes on selected values from -0.2 to 0.2 and also can be modified by the following equation:

$$A^* = A \left[1 - AKON \left(\frac{T - T_{mid}}{T_{mid} + 460} \right)^2 \right] \quad (2)$$

where

T	a given temperature value
T_{mid}	a reference temperature at midrange
$AKON$	a constant, usually 0, 15, or 30

The temperature function $\Phi(T)$ is expressed in the form of a station function, which is mathematically similar to a LaGrangian interpolation (see fig. 3 and ref. 3). The stress function is composed of two parts, each of the form

33 ASTROLOY					MEGA16				
0	3	0	1						
5	5	3	5	1					
1400	1500	1600	1700	1800					
1400	1500	1600	1700	1800					
1400	1500	1600	1700	1800					
1.16	1.632	0.04							
33	33	99000	-0.15		FTFTFTFTFTFTFTFT				
100	80	60	50	40	30	20	10	1.0	-1.0
1400	101.	12.8	0						
1400	86.	59.	0						
1400	80.	176.6	0						
1400	74.	400.7	0						
1400	70.	577.	0						
1400	61.	2279.8	0						
1400	55.	4063.2	0						
1500	75.	30.5	0						
1500	64.	142.2	0						
1500	56.	351.3	0						
1500	52.	712.	0						
1500	45.	1228.3	0						
1500	39.	2227.4	0						
1500	31.	4393.4	0						
1600	64.	10.5	0						
1600	56.5	28.8	0						
1600	46.5	145.8	0						
1600	41.	253.0	0						
1600	37.	535.7	0						
1600	31.	888.	0						
1600	24.5	2899.7	0						
1600	19.	6331.	0						
1700	41.	11.5	0						
1700	33.5	44.2	0						
1700	29.	120.9	0						
1700	24.	342.7	0						
1700	21.	746.7	0						
1700	17.5	1768.7	0						
1700	14.2	2838.7	0						
1800	29.5	6.1	0						
1800	20.5	49.3	0						
1800	17.	174.	0						
1800	10.5	340.7	0						

total of NODU values

VARIABLE NAMES	FORMAT
TITEL	16A4
NF NP NG MK	16I5
NL NT NS NISO NXAV	16I5
CISDT	16F5.0
ALOGT=TSTA	16F5.0
TSTA	16F5.0
SSTA	16F5.0
NODU NODT CUTOF AEXT OPT	2I5,F9.1,F6.3,4X,17L1
CONSTR (fields 2&10 = ALPHA)	16F5.0
TMP SIG AT WF BATCH	F5.0,F7.3,F8.2,2F5.0

COLUMN	VARIABLE	UNITS
1	temperature	degrees F
2	stress	KSI
3	time	hours
4	weighting	

Figure 2. - Typical data set and input format for MEGA16.

$$S(\log S) = D + E \log S + F S^{\pm \alpha} \quad (3)$$

where

α an integer, usually +1 for $\sigma \leq \sigma_0$ and -1 for $\sigma > \sigma_0$

The two parts are connected at a *spline point* where the second derivatives with respect to $\log S$ are forced to be equal and will also be continuous if $\alpha = -\alpha$.

The calculations of MEGA16 begin with the expression of each of the input data points in this form of the model equation

$$\begin{aligned} \log t_i = \mathcal{P}(T) [A^* \log t_i + 1] + D + E \log \left(\frac{\sigma_i}{\sigma_0} \right) + F_1 \\ \times \left[\frac{(\sigma_i/\sigma_0)^\alpha}{5.30\alpha^2} - \frac{1}{5.30\alpha^2} - \frac{\log(\sigma_i/\sigma_0)}{2.30\alpha} \right] \\ + F_2 \left[\frac{(\sigma_i/\sigma_0)^\alpha}{5.30\alpha^2} - \frac{1}{5.30\alpha^2} - \frac{\log(\sigma_i/\sigma_0)}{2.30(-\alpha)} \right] \quad (4) \end{aligned}$$

where

F_1 is set equal to 0 for $\sigma > \sigma_0$
 F_2 is set equal to 0 $\sigma \leq \sigma_0$

For the number of data used, $i = \text{NODU}$, and the model equation, a matrix of NODU by (NPAR+1) dimensions is formed and then inverted to provide estimates for the coefficients D , E , F_1 , and F_2 , as well as the station function values representing the NP discrete temperatures. Figure 4 shows, for a set of data with unevenly spaced test temperatures, an example of this input matrix. The solution, by MEGA16, of this set of equations representing the iron-nickel alloy A-286 (ref. 1) gave these coefficients: $p_i = -2.17, -0.81, 0.0, 1.39, 2.64$; $D = 3.79$; $E = -4.33$; $F_1 = 2.25$; and $F_2 = -61.41$. Predicted values of the logarithm of time to rupture are made by MEGA16 using the coefficients with combinations of temperature and stress increments. To provide the predicted isothermals, equation (4) is rewritten as

$$\log t_{est} = \mathcal{G}_{est} + \mathcal{P}_{est} \quad (5)$$

where

$$\mathcal{P}_{est} = p_1 T_1 + p_2 T_2 + \dots + p_{NP} T_{NP} \quad (6)$$

$$P(t) = P_{n-1} K_{n-1} + P_n K_n + P_{n+1} K_{n+1} + P_{n+2} K_{n+2}$$

where

$$K_{n-1} = \frac{1}{2} \frac{(t - T_n)(T - T_{n+1})}{(T_n - T_{n-1})(T_{n+1} - T_{n-1})}$$

$$K_n = \frac{1}{2} \frac{(T - T_{n-1})(T - T_{n+1}) + (T - T_{n+1})(T - T_{n+2})(T - T_n)}{(T_n - T_{n-1})(T_{n+1} - T_{n-1})}$$

$$K_{n+1} = \frac{1}{2} \frac{(T - T_{n-1})(T - T_{n+1}) + (T - T_{n+1})(T - T_{n+2})(T - T_n)}{(T_n - T_{n+1})(T_{n+1} - T_{n+2})}$$

$$K_{n+2} = \frac{1}{2} \frac{(T - T_n)(T - T_{n+1})}{(T_n - T_{n+2})(T_{n+1} - T_{n+2})}$$

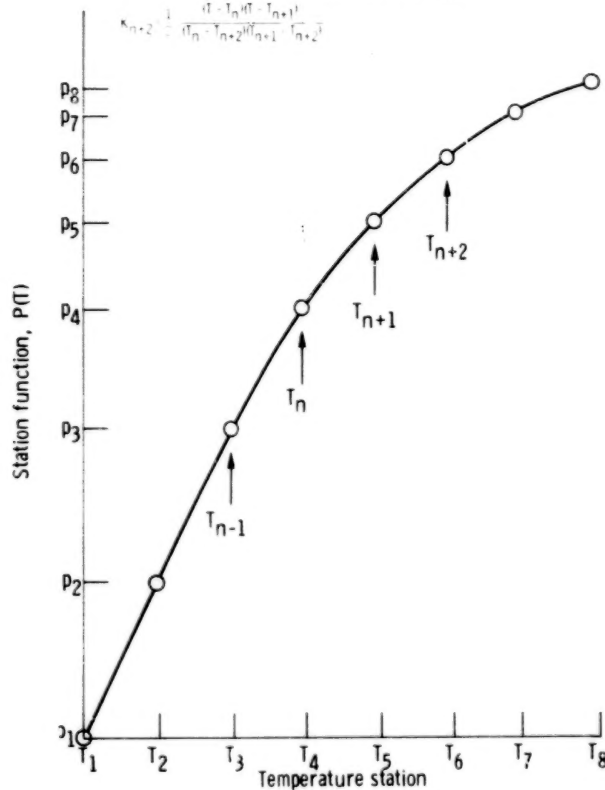


Figure 3. - Idealized temperature function, showing T_j , the discrete values of temperature stations, and corresponding p_j , or station function values.

$$G_{est} = D + E + \log\left(\frac{\sigma_i}{\sigma_o}\right)$$

$$+ F \left[\frac{(\sigma_i/\sigma_o)^\alpha}{5.30\alpha^2} - \frac{1}{5.30\alpha^2} - \frac{\log(\sigma_i/\sigma_o)}{2.30\alpha} \right] \quad (7)$$

This permits $\log t_{est}$ values to be calculated for evenly spaced values of $\log(\sigma_i/\sigma_o)$, which are then plotted versus \log stress.

Another important part of the calculation is the estimation of error. The standard deviation of the data used gives an idea of how well the model equation fits over the range of the input variables. It is calculated using

Standard deviation -

NODU

$$\frac{\sum_{i=1}^{NODU} (\log t_i - \log \hat{t}_i)^2}{NODU - NPAR - 1} \quad (8)$$

where

$\log t_i$ logarithm of observed life values

$\log \hat{t}_i$ logarithm of predicted life values

NODU number of data used

NPAR number of parameters

Output

Figure 5 gives examples of the long and short tabular outputs from a run using the input data from figure 2. Figures 6 shows some of the 16 frames that were plotted on microfiche using an adaptation of an IBM system of plotting (ref. 5). Five plots for each of three A values plus the one A scan were requested by the series of F 's and T 's (F gives a plot, T omits it) in the eighth line of input for this material (fig. 2). The first two A values (0 and -0.5) were specified by a statement in the main program of MEGA16, and the third A (-0.15) was specified in the data set (also line 8 of fig. 2).

Experience has shown that analysis of a given material is seldom, if ever, completed by the first run of that data set. This, coupled with the fact that constant A in the model equation of MEGA16 needs to be optimized, leads to the dual system of selecting trial A values; that is, a few values over the range of possible A values (-0.2 to 0.2) are set in the beginning of the main program as starting values; 0.1 and -0.05 are useful for a new set of data. A plot of the standard deviation values versus a few A values often indicates a U-shaped pattern, pointing to a possible minimum standard deviation at another A value. This other A value can then be the one read in with the data as the "extra A " value (line 8 of fig. 2 and -0.15 above) for succeeding runs. The short form of the tabular output lists only the standard deviation results for specified A values and thus is used without any other output as a quick means of finding the best A value.

Computer Program Details

MEGA16 consists of a main program of 1300 lines and subroutines totaling 500 lines, including many for documentation. For a typical single set of data on an IBM 370 time sharing system, 15 seconds of central processing time are required, but this drops

Matrix column number									
Row	T_i σ_f	σ_i ksi	t_i hr	$\log t$	k_1	k_2	k_4	k_5	g_1
1	1050	76	81.4	1.911	0.904	0	0	0	1.0
2		72	97.9	1.991	.900				.297
3		69	210.2	2.323	.834				.279
4		67	523.3	2.719	.864				.266
5		56	12833.6	4.108	.795				.188
6	1100	68	352.1	2.547	.291	0.873			.273
7	1150	65	41.9	1.622	0	.919			.253
8		62	177.7	2.250		.888			.232
9		54	644.4	2.809		.860			.172
10		43	6752.8	3.829		.808			.073
11		38	15460.8	4.189		.791			.020
12	1200	60	18.9	1.276	0				.218
13		56	154.1	2.188					.188
14		54	385.6	2.586					.172
15		47	839.0	2.924					.112
16		35	6882.4	3.838					-.016
17		29	17826.5	4.251					-.098
18	1300	40	146.9	2.167		0.892			.042
19	1300	37	335.2	2.525		.874			.008
20	1300	30	1245.7	3.095		.845			-.083
21	1350	30	282.4	2.451		.658	0.329		-.083
22	1400	30	49.3	1.693		0	.915		-.083
23	1400	22	339.0	2.530		0	.873		-.217
24	1400	15	2287.2	3.359		0	.832		-.384

$$k = P(T)(A \cdot \log t + 1) - P(T) \cdot A \cdot \log t + P(T) \text{ since AKON} = 0$$

$$g_2 = \log \log \sigma_0 \quad g_{3,4} = \left[\frac{(\log \sigma_0)^2}{\sigma^2 \ln 10^2} - \frac{1}{\sigma^2 \ln 10^2} - \frac{\log \log \sigma_0}{\sigma \ln 10} \right]$$

Figure 4. - Example of input matrix for MEGA16. Material is A-286 steel (ref. 1). Value of $A \approx -0.05$; AKON = 0; spline point = 1.56. Temperature stations are 1050°, 1150°, 1200°, 1300°, and 1400° F.

RESULTS FROM MEGA16											
LOGH : A LOGH * P(T) : P(T) = G(SIGMA)											
THE VALUES OF A	=	0.000	0.050	0.150	33	ASTROLOY	MEGA16				
S. D. OF DATA USED	=	0.121	0.107	0.079							
(a) Short printout.											
RESULTS FROM MEGA16											
LOGH : A LOGH * P(T) : P(T) = G(SIGMA)											
G STATIONS	1.160	1.630	2.004								
P STATIONS	1400.000	1500.000	1600.000	1700.000	1800.000						
G COEFFICIENTS	2.373	4.327	-0.786	-46.084							
P STATION FUNCTION	-3.204	-1.582	0.000	1.494	2.460						
DATA USED IN CALCULATION											
LOG TIME	TEMP	LOG STRESS	G	P							
085	FREQ	DIFF	SUM SG								
1.10721	1.10545	0.00176	0.00000	1400.0	2.00432	-1.71770	-3.20422				
1.77085	1.09317	-0.12232	0.01496	1400.0	1.93450	0.65844	-3.20422				
2.24699	2.22124	0.02575	0.01563	1400.0	1.90309	0.21727	-3.20422				
2.60282	2.55466	0.04815	0.01795	1400.0	1.86923	0.23109	-3.20422				
2.76118	2.77860	-0.01742	0.01825	1400.0	1.84510	0.53222	-3.20422				
3.35790	3.27931	0.07858	0.02443	1400.0	1.78533	1.20554	-3.20422				
3.60887	3.60017	0.00870	0.02450	1400.0	1.74036	1.63700	-3.20422				
1.48430	1.42398	0.06032	0.02814	1500.0	1.87506	0.15600	-1.58200				
2.15290	2.10151	0.05139	0.03078	1500.0	1.80618	0.98295	-1.58200				
2.54568	2.57996	-0.03428	0.03196	1500.0	1.74819	1.56691	-1.58200				
2.85248	2.80454	0.04794	0.03425	1500.0	1.71600	1.84101	-1.58200				
3.08930	3.14803	-0.05872	0.03770	1500.0	1.65321	2.26024	-1.58200				
3.34780	3.37789	-0.03009	0.03861	1500.0	1.59106	2.54079	-1.58200				
3.64280	3.72627	-0.08347	0.04557	1500.0	1.49136	2.96600	-1.58200				
1.02119	0.98295	0.03824	0.04704	1600.0	1.80618	0.98295	0.00000				
1.45939	1.53154	-0.07214	0.05224	1600.0	1.75205	1.53154	0.00000				
2.16376	2.17941	-0.01565	0.05249	1600.0	1.66745	2.17941	0.00000				
2.40312	2.44727	-0.04415	0.05444	1600.0	1.61278	2.44727	0.00000				
2.72892	2.63887	0.09005	0.06254	1600.0	1.56820	2.63887	0.00000				
2.94841	2.94600	-0.01759	0.06285	1600.0	1.49136	2.94600	0.00000				
3.46235	3.39594	0.06642	0.06726	1600.0	1.38917	3.39594	0.00000				
3.80147	3.85515	-0.05367	0.07015	1600.0	1.27875	3.85515	0.00000				
1.06070	1.20350	0.14280	0.09054	1700.0	1.61278	2.44727	1.49449				
1.64542	1.67817	-0.03275	0.09161	1700.0	1.52504	2.82306	1.49449				
2.08243	2.01334	0.06909	0.09438	1700.0	1.46240	3.08840	1.49449				
2.53491	2.44908	0.08583	0.10375	1700.0	1.38000	3.43337	1.49449				
2.87315	2.75423	0.11892	0.11789	1700.0	1.32222	3.67494	1.49449				
3.24765	3.16826	0.07940	0.12419	1700.0	1.24304	4.00272	1.49449				
3.45312	3.59262	0.13950	0.14365	1700.0	1.16137	4.33868	1.49449				
0.70533	0.81155	0.02622	0.14424	1800.0	1.46982	3.05707	2.46033				
1.69285	1.71094	0.01809	0.14467	1800.0	1.31175	3.71040	2.46033				
2.24055	2.16821	0.07234	0.14990	1800.0	1.23045	4.05464	2.46033				
2.53237	2.55450	0.02212	0.15039	1800.0	1.16137	4.33868	2.46033				
33	ASTROLOY	MEGA16									
S. D. OF RES	=	0.079	DOF	=	24.0	SUM OF RES	=	0.012	AV ABS DEV	=	0.057
THE VALUE OF 'A' = 0.150											

(b) Long printout.

Figure 5. - Example of output from MEGA16.

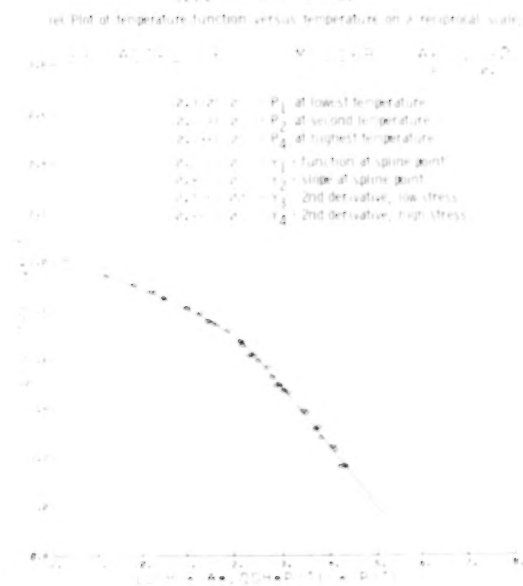
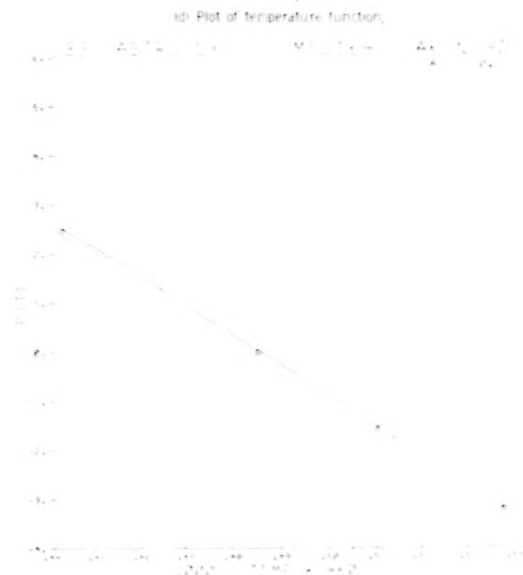
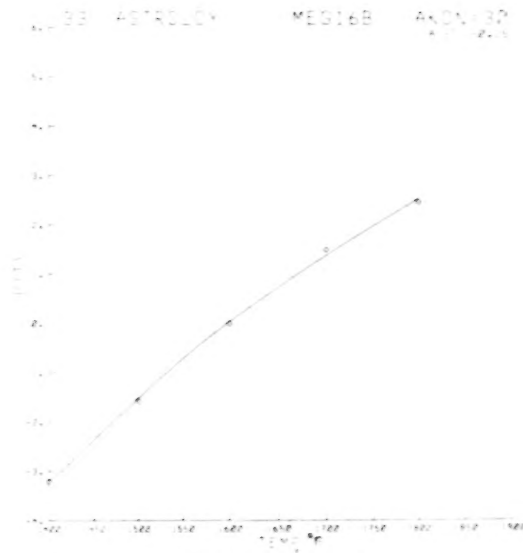
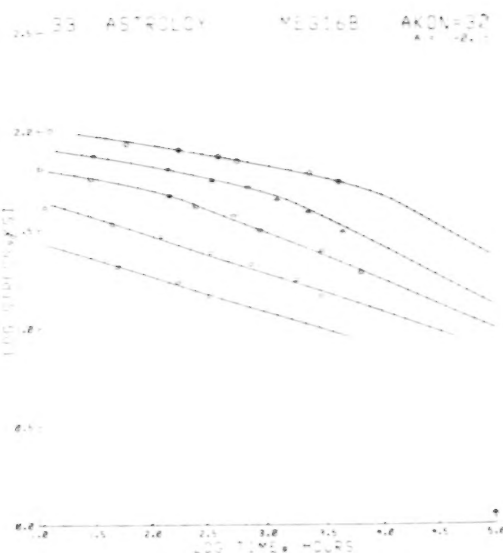
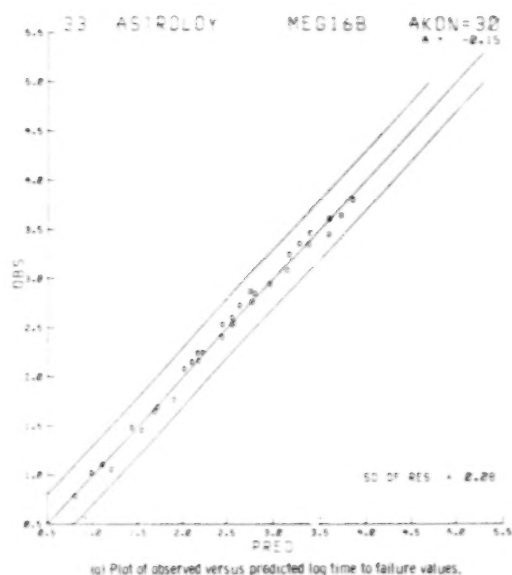
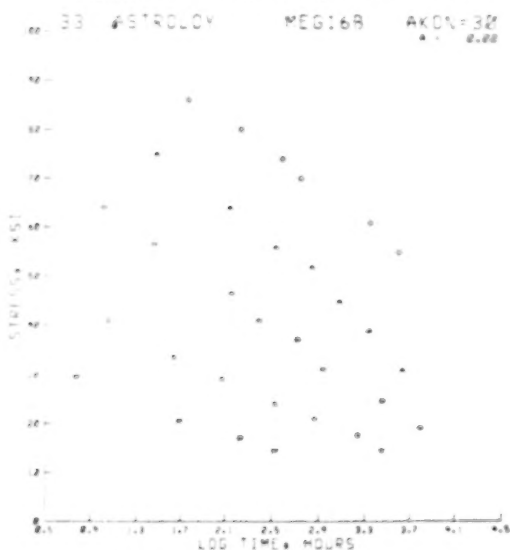


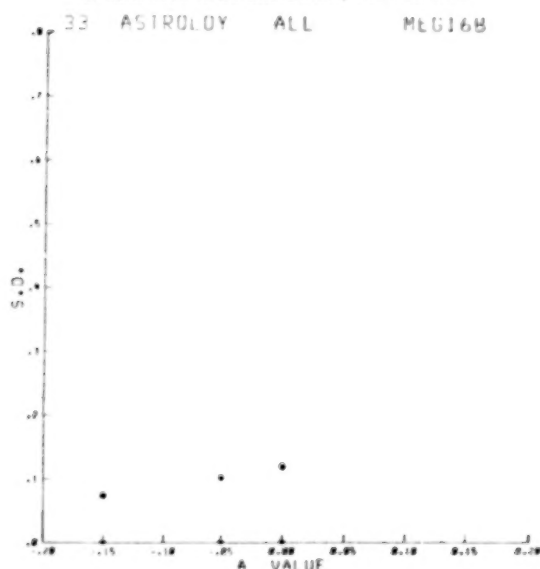
Figure 6. Sample plots.



(i) Plot of observed versus predicted log time to failure values.



(ii) Plot of raw data for preassessment using linear stress scale.



(iii) The A scan: observed standard deviation versus selected A value.
Figure 6. - Concluded.

rapidly to about 5 seconds per data set, if many sets are run consecutively after the program is loaded. When run on this virtual memory computer MEGA16 uses approximately 2800 K of eight bit bytes.

Discussion

The question of accuracy, somewhat exemplified by the observed versus predicted plot in each run of MEGA16, has been discussed (refs. 1 to 3), but not really answered. A point not always considered is that the accuracy of a prediction cannot be much better than the precision of the data used to make that prediction. Of course the scatter of a given set of creep-life data depends on many unquantified variables in the general categories of test procedure, metallurgical differences, environment, etc. Estimates of the extent or importance of this scatter are not often presented with creep-life data. Few researchers are inclined to run the expensive replicate tests necessary to given meaningful estimates of the experimental error for various materials and laboratory conditions. To provide a rough estimate of scatter to be found in creep-life experiments, the data analyzed in the development of MEGA16 (ref. 4) were also used to provide the measures of average deviations in table I. Shown are some statistics for the temperature and stress combinations which had three or more repeat tests. Using the usual assumption of a lognormal distribution of time to failure, the last column of this table shows a fairly wide variation in the scatter about the mean log life for various combinations of material temperature and stress. For some materials standard deviations of 0.1 appear to be typical, but for others 0.2 or over are to be expected. Therefore, it is reasonable to assume that the best accuracy to be achieved from a predictive technique will also be somewhere in the range of 0.1 of a decade of the logarithm of the time to rupture, but there will be cases of higher deviation for certain materials and testing conditions.

With the stress function of MEGA16, there is the need for choosing an intermediate value of the logarithm of stress to be used as the spline point. Normally, a value somewhere near the midrange of stress will suffice, but for some data selection of another log stress value will change the fit and give different extrapolations. As an example, some additional calculations were made using the type 316 stainless steel set of data used in reference 4. In that report the location of the spline point for this material had been, for other reasons, selected at a point somewhat distant from the midrange. Figures 7 and 8 show the isothermals and master curves for some of the selected combinations of A and spline point locations. Table II gives the resulting standard

TABLE I.—STATISTICS OF REPEAT MEASUREMENTS

TEMP F	STRESS KSI	NO. OF REPEATS	LIFE		LOG LIFE	
			MEAN (a)	C.V. (b)	MEAN (a)	S.D. (c)
64 1100-O ALUMINUM						
482.	6.0	3	299.67	41.63	2.445	0.212
482.	5.5	3	1274.67	58.67	3.055	0.258
662.	4.0	3	49.17	7.22	1.691	0.031
662.	3.5	4	203.75	14.28	2.306	0.061
752.	3.5	6	13.08	37.86	1.097	0.136
752.	3.0	7	62.71	15.99	1.793	0.070
932.	1.5	3	224.33	26.17	2.340	<u>0.122</u>
						0.127
75 5454-O ALUMINUM						
662.	14.0	3	81.67	26.67	1.902	0.112
662.	11.0	5	436.00	14.32	2.636	0.063
752.	9.0	7	165.71	13.50	2.216	0.059
842.	7.0	4	96.50	9.48	1.993	0.041
932.	5.0	3	113.00	13.74	2.050	<u>0.060</u>
						0.067
95 INCO 625						
1200.	70.0	6	46.57	83.72	1.501	0.476
1200.	55.0	3	769.53	69.57	2.791	0.409
1300.	50.0	4	46.67	80.86	1.549	0.382
1300.	45.0	7	246.94	54.79	2.307	0.336
1400.	30.0	5	267.52	69.10	2.326	0.350
1500.	20.0	5	43.18	32.88	1.617	0.139
1500.	17.5	5	87.66	22.75	1.933	0.107
1500.	15.0	9	506.80	73.06	2.601	0.321
1500.	12.5	4	1722.42	53.93	3.171	0.298
1500.	10.0	5	7117.70	34.85	3.834	0.139
1600.	12.0	3	44.00	19.38	1.638	<u>0.082</u>
						0.276
105 U-500						
1200.	140.0	6	7.53	35.93	0.846	0.192
1200.	100.0	8	1092.57	57.37	2.981	0.239
1500.	60.0	8	17.99	31.58	1.235	0.141
1500.	30.0	6	1060.93	50.99	2.959	0.288
1800.	9.0	3	83.57	59.33	1.849	<u>0.337</u>
						0.239
104 L-605						
1200.	60.0	8	7.56	38.03	0.851	0.166
1200.	40.0	6	1129.18	30.59	3.035	0.140
1500.	30.0	8	14.45	28.42	1.147	0.111
1500.	17.0	7	1618.47	19.68	3.201	<u>0.092</u>
						0.127
101 6061-T6 ALUMINUM						
662.	26.0	3	173.00	18.05	2.234	0.076
752.	26.0	3	28.37	133.22	1.163	0.602
752.	21.0	6	76.67	20.27	1.878	0.077
842.	13.0	4	184.38	30.40	2.250	0.134
842.	11.0	3	751.33	22.10	2.869	0.092
932.	13.0	4	19.30	59.09	1.225	0.267
932.	11.0	3	74.33	12.50	1.869	0.056
1112.	4.0	3	133.33	7.09	2.124	<u>0.030</u>
						0.167

$$^a \text{MEAN} = \frac{\sum x}{n}$$

$$^b \text{C.V.} = \frac{\sqrt{\text{VAR}}}{\text{MEAN}} \times 100$$

$$^c \text{S.D.} = \sqrt{\text{VAR}} \text{ where } \text{VAR} = \frac{\sum x^2 - (\sum x)^2 / n}{n - 1}$$

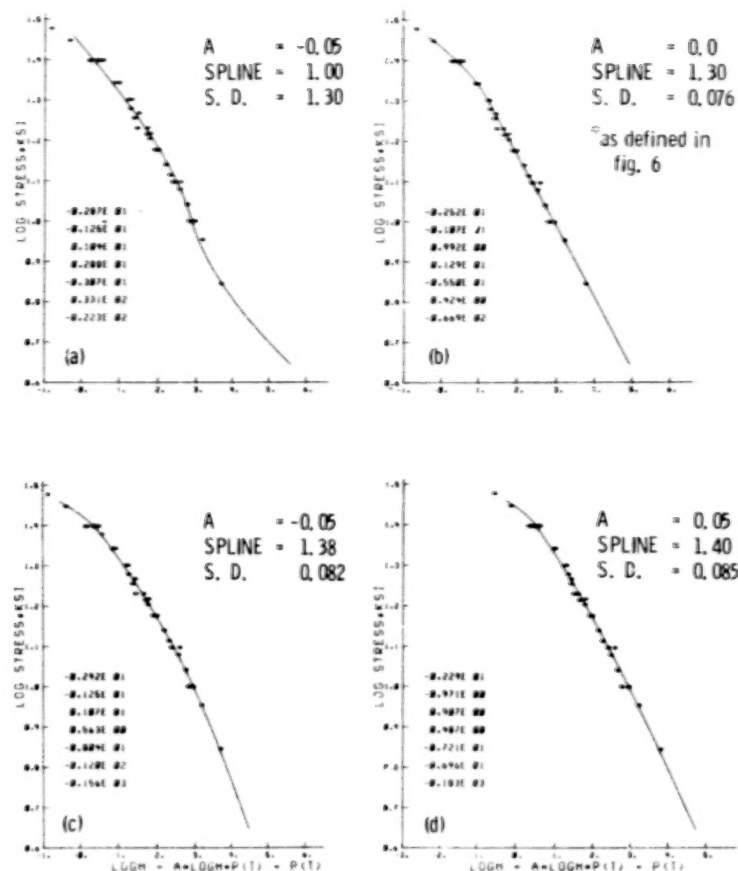


Figure 8. - Results from MEGA16 showing effect of different A values and spline points on shape of master curves. Material, 316 stainless steel.

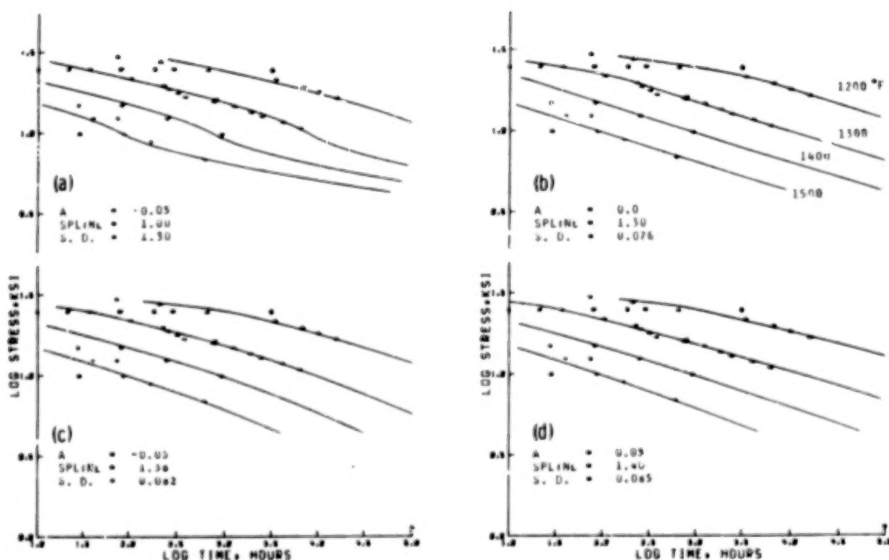


Figure 7. - Results from MEGA16 showing effect of different A values and spline points on shape of predicted isothermals. Material, 316 stainless steel.

TABLE II.—EFFECT OF LOCATION OF SPLINE POINT IN THE STRESS FUNCTION ON THE STANDARD DEVIATION OF PREDICTIONS

An example using type 316 stainless steel with 38 data and a stress range of 7 to 30 ksi. The program is MEGA16 with AKON = 30.

Location of spline point	A value		
	-0.05	0	0.05
Standard deviation			
1.0	0.130	0.125	0.125
1.1	0.123	.118	.116
1.2	0.109	.103	.102
1.3	0.082	.076	.076
1.38	0.082	.080	.082
1.4	0.085	.083	.085

deviations for all the combinations. From the table, it appears that a minimum standard deviation exists near 0.076, which is a lower value than that reported in figure 39 of reference 4. Thus, choice of spline point location might be a means of fine tuning the predictions of MEGA16, but from figure 7(a) it is obvious that extreme values of the spine point may give strange predicted isothermals.

Summary of Results

MEGA16 is the outcome of a number of years of study of various forms of the so-called minimum commitment method of analysis of stress-rupture data. Basically, it is a Fortran IV program which fits a specific, yet flexible, model equation to a given family of time-to-failure data. The model includes a discrete temperature function, a two-part stress function, and an adjustable constant, A , which affects the extrapolation beyond the range of data. MEGA16 can handle up to 200 data points per set, including the weighting of certain points, if so desired. For each datum it sets up an equation according to the assumed model and some optional locations of the centers of the temperature and stress functions. After it solves the resulting set of equations to determine the coefficients of the terms

of the model equation, it predicts the rupture life for each input combination of temperature and stress and predicts the logarithm of the time of evenly spaced values of the logarithm of stress.

MEGA16 produces many different types of plots to depict the response of a particular material. These include (1) the basic predicted time-to-rupture versus stress for specific temperatures, (2) the temperature and stress functions, (3) the time and temperature function versus the logarithm of stress (the master curve), (4) the observed versus predicted log time to failure values, (5) residual plots of the logarithm of time, stress, and temperature, and (6) standard deviation as a function of the parameter A . MEGA16 calculates, as an estimate of accuracy, the standard deviation of the observed minus the predicted rupture life.

To be analyzed by MEGA16, the data set representing a material should include test results from at least three different temperatures, each with several levels of stress, and the resulting spread of life values. That set can be run alone or with many other sets of data from other materials to obtain complete analyses, in easy-to-understand graphical form.

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio, August 28, 1980

Appendix—Listing of MEGA16 and its Subroutines

THIS IS MEGA16 ----THE MANSON-ENSIGN GENERALIZED ANALYSIS
OF STRESS RUPTURE DATA----INCORPORATING THE IDEAS OF MINIMUM
COMMITMENT TO FORM OF EQUATION, THE STATION FUNCTION APPROACH,
AND A MODEL EQUATION OF THE FORM

$$\text{LOGH} + (1 + A \text{ LOGH})P(T) = G(S)$$

WHERE H = TIME

P = STATION FUNCTION FOR TEMPERATURE

T = TEMP

G = STRESS FUNCTION, $D + E \text{ LOG } S + F S^{**}ALPHA$

ALPHA = +1 OR -1

S = STRESS / REFERENCE STRESS

A = A CONSTANT USUALLY BETWEEN -.2 AND +.2

OR $A = A(1 - ACON(T - TMID / TMID + 460))^{**2}$

A S S U M P T I O N S

FORM OF THE MODEL EQUATION

STRESS AND TEMPERATURE ARE THE ONLY
INDEPENDENT VARIABLES

RELATIONSHIPS AMONG STRESS RUPTURE LIFE,
STRESS, AND TEMP PRESENT IN THE SHORT
TIME OBSERVATIONS ALSO EXIST IN
THE LONG TIME BEHAVIOR

CALL STRUCTURE

	TUSTR		
	STACH		
MEG16B	AMATF	DMFSS	
		DMLSS	
	TOPLT	CVFITF	DTRIA
			DSOLVE
			FUNC

THE SUBROUTINES CALLED BY THIS PROGRAM (MEG16B) ARE

NAME	SOURCE	PURPOSE
TUSTR	STUSTR	CONVERTS LOG STRESS VALUE TO STATION FUNCTION FORM
STACH	SSTACH	CONVERTS TEMP VALUE TO STATION FUNCTION FORM
AMATF	SAMATF	SETS UP THE MATRIX OF STATION FUNCTION EQUATIONS(UMATRX)
TOPLT	STOPLT	PREPARES STATION FUNCTION VALUES (XF) FOR PLOTS AND CURVE FITS

OTHER SUBROUTINES CALLED ARE

IN	NAME	SOURCE	PURPOSE
	AMATF	DMFSS	SDMF PREPARES MATRIX, FINDS RANK, AND LINEARLY IND. ROWS
	AMATF	DMLSS	SDML INVERTS UMATRX TO FIND LEAST SQUARES SOLUTION (XF)
	TOPLT	CVFITF	CRVFF POLYNOMIAL CURVE FIT -- CROUSE-JORDAN REDUCTION
	CVFITF	DIRIA	DCTRIA
	CVFITF	DSOLVE	DCSOLV
	CVFITF	FUNC	FUNCT

ADDITIONAL SUBROUTINES CALLED ARE FROM THE IBM PLOTTING PACKAGE

THESE INCLUDE GLOT, XAXIS, YAXIS, CHARS, NUMBER, BEGID, TITLE, SCLBAK, ETC.

NOMENCLATURE ***** INPUT VALUES

TITEL IS THE TITLE FOR A PARTICULAR MATERIAL
 NF DENOTES THE F COLUMN TO BE OMITTED IN FINAL MATRIX
 NP DENOTES THE P COLUMN TO BE OMITTED IN FINAL MATRIX
 NG DENOTES THE G COLUMN TO BE OMITTED IN FINAL MATRIX
 MK DENOTES THE COLUMN TO BE USED AS THE DEPENDENT VARIABLE
 NL IS THE NUMBER OF STATIONS FOR TIME--NOT USED IN THIS VERSION--SET = NT
 NT IS THE NUMBER OF STATIONS FOR TEMP MAXIMUM = 32
 NS IS THE NUMBER OF STATIONS FOR STRESS MAXIMUM = 16
 NO IS THE NUMBER OF STRESS PARAMETERS BEING DETERMINED
 NCOL IS THE NUMBER OF COLUMNS IN UMATRX = NUMBER OF BOTH
 DEPENDENT PLUS INDEPENDENT VARIABLES MAXIMUM = 48
 NEV IS THE NUMBER OF EVENLY SPACED LOG TIME VALUES AT
 WHICH PREDICTIONS ARE MADE
 NISO IS THE NUMBER OF ISOTHERMALS TO BE PREDICTED
 WHICH ARE NOT ALSO STATION FUNCTIONS
 NXAV IS NUMBER OF EXTRA A VALUES READ IN (MAX=1 FOR NOW)
 CISOT(K) ARE THE TEMP VALUES OF THE ISOTHERMALS
 ALOGT ARE THE LOG TIME STATIONS (NOT USED IN THIS VERSION)
 TSTA ARE THE TEMP STATIONS
 SSTA ARE THE LOG STRESS STATIONS
 NODU IS THE NUMBER OF INPUT DATA POINTS -- DATA USED
 NODUW IS THE NUMBER OF DATA USED AFTER WEIGHTING
 NODP IS THE NUMBER OF LONG TIME DATA TO BE PREDICTED (NOT IN THIS VERSION)
 NOEXT IS THE NUMBER OF DATA ADDED BY THE WEIGHTING
 CUTOF IS THE CUTOFF LIFE IN HOURS
 AEXT IS THE EXTRA A VALUE READ IN WITH DATA
 OPT IS THE PLOT OUTPUT OPTION; F (OR 0) MEANS PLOT, T (1) MEANS NO PLOT
 CONST ARE THE VALUES OF CONSTANT STRESS FOR THE
 ISOSTRESS PLOTS
 TMP(I) ARE THE TEMP VALUES OF THE DATA IN DEGREES FAHREHEIT
 SIG(I) ARE THE STRESS VALUES OF THE DATA IN KSI
 ALS(I) ARE THE LOG STRESS VALUES
 AT(I) ARE THE TIME TO FAILURE VALUES OF THE DATA IN HOURS
 ALT(I) ARE THE LOG TIME VALUES
 WF IS THE NUMBER OF TIMES A POINT IS TO BE WEIGHTED


```

C      COMMON /ACOMN/ QXPLTT(32),QXPLRT(32),QYPLTT(32),QXPLTG(16),QYPLTG(16)
COMMON /BCOMN/ XPLTT(32),TFCN(32),XPLRT(32),XPLTG(16),YPLTG(16)
COMMON /CCOMN/ XF(48),NP
COMMON /DCOMN/ D,E,F,G,H,AA

C
C
C      PRINT OUT TO THIS DATE ON EACH PLOT
      INTEGER*2 NDATE

C      DIMENSION DATES(4)
      DATA NDATE /16/
      DATA DATES(1),DATES(2) / 'DATE',' -- ' /
      DATA DATES(3),DATES(4) / 'MO/D','D/YY' /
      CALL TIME (NDATE,DATES)
      WRITE (7,516) (DATES(I), I=1,4)

C
      DATA QLABX1 / ' LOG',' TIM','E, H','OURS' /
      DATA QLABY1 / ' LOG',' STR','ESS',' KSI ' /
      DATA QLABY2 / 'G(S)' /
      DATA QLABX2 / ' LOG ',' STRE','SS ' /
      DATA QLABX3 / 'TEMP' /
      DATA QLABY3 / 'P(T)' /
      DATA QLABY4 / 'RESI','DUAL','S O','F L','OG T' /
      DATA QLABX5 / 'LOGH',' + A','*LOG','H*P(','T) +',' P(T',')' /
      DATA QLABX7 / 'PRED' /
      DATA QLABY7 / ' OBS' /
      DATA QLABX8 / 'STRE','SS, ',' KSI' /
      DATA QLABX4 / ' ','1000',' / T','EMP ','+ 46','0 ' /
      DATA QLABXA / 'A V','ALUE' /
      DATA QLABYA / 'S.D.',' OF ','PRED','ICTI','ONS ' /
      DATA QLABXD / 'LOG ','TIME','A =',' 0 ' /
      DATA QLABYD / 'LOG ','TIME','A =',' X ' /
      DATA QLABXC / 'LOG ','T, A','=0 ','- LO','G CU','TOFF',' ' /
      DATA QLABYC / 'LOG ','T, A','=X ','- LO','G T',' A=0',' ' /

C
      DATA KSYM /70,66,62,65,68,56,67,61,72,92,78,69,58,63,71, -
1 70,66,62,65,92,63,67,61,72,78,69,71,56,143,181 /
      DATA NEV /34/, ALTT / 1.0,1.125,1.25,1.375,1.5, -
11.625,1.75,1.875,2.0,2.125,2.25,2.375,2.5,2.625,2.75,2.875,3.0, -
23.125,3.25,3.375,3.5,3.625,3.75,3.875,4.0,4.125,4.25,4.375,4.5, -
34.625,4.69897,4.75,4.875,5.0/

C
C      NOTA BENE
C
C      THE FIRST A SHOULD BE ZERO TO INSURE THAT ALL PLOTS HAVE MEANING
C      SET LAK TO AT LEAST 1 LESS THAN DIMENSIONS OF VALUA
C      DATA LAK /5/, VALUA /0.0,-.15,-0.1,-0.05,.15/
      DATA LAK /2/, VALUA /0.00,-0.05/

C
      DATA GRLEG5 / ' A =',' ' /
      DATA QPX /0.5,5.3/
      DATA QPY /0.5,5.3/
      DATA QSDKEY / 'SD O','F RE','S =',' ' /
      DATA QEXKEY / 'RMS ',' OF P','RED=',' ' /
      DATA QAVKEY / 'AV D','EV P','RED=',' ' /
      DATA QABKEY / 'AB D','EV P','RED=',' ' /
      DATA QADKEY / 'AV A','BS D','EV =',' ' /
      DATA QRDKEY / 'SD R','EDUC','ED =',' ' /

C
      DATA QPXL /0.5,4.699/
      DATA QPYL /0.801,5.0/
      DATA QPXR /0.801,5.301/
      DATA QPYR /0.5,5.0/
      DATA QPX2 /2.0,5.0/
      DATA QPY2 /2.0,5.0/
      DATA QPXL2 /2.0,4.699/
      DATA QPYL2 /2.301,5.0/
      DATA QPXR2 /2.301,5.0/
      DATA QPYR2 /2.0,4.699/

```

```

C
C  INITIALIZE--BEGIN WITH A NEW DATA SET
C
100  DO 105 I=1,200
      TMP(I)=0.0
      SIG(I)=0.0
      AT(I)=0.0
105  WF(I)=0.0
      DO 106 I=1,9
          PSTDEV(I)=0.0
          PSDEV(I)=0.0
          PSTDL(I)=0.0
106  PSDZD(I)=0.0
      STDEV = 0.0
      DO 107 J=1,200
          ALT(J) = 0.0
          ALS(J) = 0.0
107  CONTINUE
      READ (5,410) TITEL

C
C  INPUT--STATION VALUES AND CONSTANTS
C
      READ (5,402) NF,NP,NG,MK
      READ (5,402) NL,NT,NS,NISO,NXAV
      READ (5,400) (CISOT(K),K=1,NISO)
      READ (5,400) (ALOGT(I),I=1,NL)
      READ (5,400) (TSTA(I),I=1,NT)
      READ (5,400) (SSTA(I),I=1,NS)
      READ (5,404) NODU,NODT,CUTOF,AEXT,(OPT(I),I=1,17)
      NCS=8
      READ (5,400) (CONSTR(K),K=1,NCS),ALPHL,ALPHH
C  WRITE (6,515)
      WRITE (7,515)
      WRITE (9,515)
C  WRITE (6,455) TITEL
      WRITE (7,455) TITEL
C  WRITE (7,513)
      WRITE (9,455) TITEL
      WRITE (9,463) (SSTA(I),I=1,NS)
      WRITE (9,467) (TSTA(I),I=1,NT)
C  WRITE (7,477) NF,NP,NG,MK,NL,NT,NS,NISO,NXAV
      KTEST=1
      CUTLOG = ALOG10(CUTOF)
      CUTHAF = (CUTLOG+5.0)/2.0
      DO 111 I=1,NS
          CANGLS(I) = 10.0**SSTA(I)
111  CONTINUE
      IF (NXAV.NE.1) GO TO 109
      LAK = LAK+NXAV
      VALUA(LAK) = AEXT
109  NOEXT=0

C
C  INPUT--OBSERVATIONS
C
      DO 115 I=1,NODT
          READ (5,408,END=399,ERR=113) TMP(I),SIG(I),AT(I),WF(I)
          ALT(I)=DLOG10(AT(I))
          ALS(I)=DLOG10(SIG(I))
          GO TO 114
113  CONTINUE

C
C  WEIGHT DATA
C
114  IF (NOEXT.GT.0) GO TO 117
      KN = I
      GO TO 118
117  KN = KN+1

```

```

118  WTMP(KN) = TMP(I)
      QTMP(KN) = TMP(I)
      WALS(KN) = ALS(I)
      QALS(KN) = ALS(I)
      WALT(KN) = ALT(I)
      QALT(KN) = ALT(I)
      IF (WF(I).LE.1.0) GO TO 116
      KWF = WF(I)-1.0
      DO 112 KK= 1,KWF
      KN = KN+1
      WTMP(KN) = TMP(I)
      WALS(KN) = ALS(I)
      QALS(KN) = ALS(I)
      WALT(KN) = ALT(I)
      NOEXT = NOEXT+1
112  CONTINUE
116  IF (I.EQ.1) GO TO 115
      IF (WTMP(I).EQ.WTMP(I-1)) GO TO 115
      KTEST=KTEST+1
115  CONTINUE
      NODUW = NODU+NOEXT
      NODTW = NODT+NOEXT

```

```

C
C  CHOOSE AN  A  VALUE
C

```

```

121  DO 385 LA=1,LAK
C

```

```

C  INITIALIZE ALL THE VECTORS IN COMMON

```

```

      DO 119 K=1,48
      XF(K) = 0.0
      XPLTT(K) = 0.0
      TFCN(K) = 0.0
      XPLTG(K) = 0.0
      YPLTG(K) = 0.0
      QXPLTT(K) = 0.0
      QYPLTT(K) = 0.0
      QXPLTG(K) = 0.0
      QYPLTG(K) = 0.0
      QXPLRT(K) = 0.0
      XPLTRT(K) = 0.0

```

```

119  CONTINUE
      DO 123 I=1,16
      SFCN(I)=0.0

```

```

123  CONTINUE
      DO 120 I=1,48
      DO 120 J=1,200

```

```

120  UMATRX(J,I)=0.0

```

```

C
C  BEGIN PLOT OF ISOTHERMALS
C

```

```

      CALL BEGID(1)
      IVGR(1) = 7
      IVGR(2) = 1
      IVGR(3) = 3
      IVGR(4) = 70
      IVGR(5) = 1
      IVGR(6) = 15
      IVGR(7) = 0
      GRAXV(1) = 10.
      GRAXV(2) = -1.
      GRAXV(3) = 0.
      GRAXV(4) = 1.
      GRAXV(5) = 5.
      GRAXV(6) = 8.
      GRAXV(7) = 2.
      GRAXV(8) = -1.
      GRAXV(9) = 0.
      GRAXV(10) = 12.
      GRAYV(1) = 10.

```



```

GRAYV(2) = -1.
GRAYV(3) = 90.
GRAYV(4) = .0
GRAYV(5) = 2.5
GRAYV(6) = 5.
GRAYV(7) = 2.
GRAYV(8) = -1.
GRAYV(9) = 0.
GRAYV(10) = 12.
CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IA=1
DO 140 K=1,KTEST
IK=0
DO 130 I=IA,NODTW
IK=IK+1
IF (I.EQ.IA) GO TO 125
IF (WTMP(I).NE.WTMP(I-1)) GO TO 135
125 QXPL(IK)=WALT(I)
QYPL(IK)=WALS(I)
130 CONTINUE
135 IA=I
IPLT=IK-1
IF (K.EQ.KTEST) IPLT=IPLT+1
C
C PLOT THE DATA FOR THE K TH TEMPERATURE
C
IVGR(2) = IPLT
IVGR(3) = 3
IVGR(4) = KSYM(K)
CALL GPLOT (QXPL,QYPL,IVGR)
140 CONTINUE
C WRITE (6,518) (IVGR(I),I=1,7)
C WRITE (6,501) (GRAXV(I),I=1,10)
C WRITE (6,501) (GRAYV(I),I=1,10)
C WRITE (6,519) ((QXPL(I),I=1,6),(QYPL(I),I=1,6))
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,16,20,QLABY1)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,16,20,QLABX1)
C
C FORM SET OF SIMULTANEOUS EQUATIONS (UMATRX)
C
NQ=4
AKON=0.0
141 NCOL=NQ+1+NT
IF (NP.GT.0) NCOL=NCOL-1
C NOT A BENE MEG16B
C
C NUMBER OF UNKNOWN IS 4 + NT - 1 NUMBER OF COLS IN UMATRX IS 4 + NT
C THIS USES S = STRESS / STRESS MID
C ALSO ALPHH = EXPONENT ON S HIGH STRESS
C ALPHL = EXPONENT ON S LOW STRESS
C
IF (ALPHH.NE.0.0) GO TO 143
ALPHH = -1.0
ALPHL = 1.0
143 DO 210 J=1,NODUW
142 SCOEFG=WALT(J)
YVALG = WALS(J)
TRIP(J)=((WTMP(J)-TSTA(NP))/(TSTA(NP)+460.))*2
VALU(J)=VALUA(LA)*(1.0-AKON*TRIP(J))
161 CALL TISTR (SSTA,YVALG,NQ,ALPHL,ALPHH,COEFG)
C WRITE (7,450) (COEFG(K),K=1,NQ)

```

```

162 KNT=0
DO 170 K=1,NT
IF (WTMP(J).NE.TSTA(K)) GO TO 165
163 PROD(J)=VALUM(J)*WALT(J)+1.0
TMPT(K)=PROD(J)
GO TO 170
165 TMPT(K)=0.0
KNT=KNT+1
170 CONTINUE
171 IF (KNT.NE.NT) GO TO 190
C
TVALP = WTMP(J)
186 CALL STACN (TSTA,TVALP,NT,TMPT,N)
PRDD(J)=VALUM(J)*WALT(J)+1.0
187 DO 193 K=1,NT
192 TMPT(K) = TMPT(K)*PRDD(J)
193 CONTINUE
C WRITE (7,445) (TMPT(K),K=1,NT)
190 CONTINUE
194 M=1
UMATRX(J,M)=SCOEF
DO 200 I=1,NT
IF (I.EQ.NP) GO TO 200
M=M+1
UMATRX(J,M)=TMPT(I)
200 CONTINUE
DO 205 I=1,NQ
M=M+1
UMATRX(J,M)=COEFG(I)
205 CONTINUE
210 CONTINUE
IEND=NODUW
225 KSPARS=0
C
C SOLVE SET OF SIMULTANEOUS EQUATIONS
C
226 CALL AMATF (NCOL,IEND,MK,KSPARS,UMATRX)
227 CALL TOPLT (TSTA,NT,SSTA,NQ,SFCN)
C WRITE (6,457) VALUA(LA)
WRITE (9,465) (SFCN(L),L=1,NQ)
WRITE (9,469) (TFCN(L),L=1,NT)
C1 = 2.302585092
C2 = 1.0/(C1*ALPHL)
C3 = 1.0/(C1*ALPHH)
C4 = C2*C2
C5 = C3*C3
UNKN(1) = SFCN(1)-C4*SFCN(3)
UNKN(2) = SFCN(2)-C2*SFCN(3)
UNKN(3) = C4*SFCN(3)
UNKN(4) = SFCN(1)-C5*SFCN(4)
UNKN(5) = SFCN(2)-C3*SFCN(4)
UNKN(6) = C5*SFCN(4)
UNKN(7) = UNKN(2)+C1*ALPHL*UNKN(3)
UNKN(8) = UNKN(3)*C1*C1*ALPHL*ALPHL
UNKN(9) = UNKN(5)+C1*ALPHH*UNKN(6)
UNKN(10) = UNKN(6)*C1*C1*ALPHH*ALPHH
C WRITE (7,524) UNKN
SDEL = (SSTA(NS)-SSTA(1)+0.2)/50.0
C ADDING .1 TO GIVE SOME EXTRAPOLATION 18 NOV 76
STRL(1) = SSTA(1)-0.2
DO 231 K = 2,50
STRL(K) = STRL(K-1) + SDEL
231 CONTINUE
KPLT = 0
DO 232 J = 1,50
ALSTRL(J) = 10.0**STRL(J)
SVAL = 10.0**((STRL(J)-SSTA(2)))
SVLG = DLOG10(SVAL)
GLOW(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(3)*(C4*SVAL**ALPHL-C4-C2*SVLG)

```

```

      GHIGH(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(4)*(C5*SVAL**ALPHH-C5-C3*SVLG)
      IF (STRL(J).GT.SSTA(2)) GO TO 229
      GPLT(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(3)*(C4*SVAL**ALPHL-C4-C2*SVLG)
      GO TO 228
229  GPLT(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(4)*(C5*SVAL**ALPHH-C5-C3*SVLG)
228  KPLT = KPLT + 1
232  CONTINUE
      KPLT = 0
      DO 233 J = 1,NODT
      SVAL = 10.0**((WALS(J)-SSTA(2)))
      SVLG = DLOG10(SVAL)
      IF (WALS(J).GT.SSTA(2)) GO TO 234
      GPLAT(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(3)*(C4*SVAL**ALPHL-C4-C2*SVLG)
      GO TO 235
234  GPLAT(J) = SFCN(1)+SFCN(2)*SVLG+SFCN(4)*(C5*SVAL**ALPHH-C5-C3*SVLG)
235  KPLT = KPLT + 1
233  CONTINUE
C
C  PLOT THE ISOTHERMALS OF INTEREST THAT ARE NOT SAME AS STATIONS
C
      IF (NISO.EQ.NT) GO TO 214
      DO 211 ISO=1,NISO
      PISO(ISO) = D+E*CISOT(ISO)+F*CISOT(ISO)**2
      KP = 0
      TRIP(ISO) = (((CISOT(ISO)-TSTA(NP))/(TSTA(NP)+460.))**2
      VALUM(ISO) = VALUA(LA)*(1.0-AKON*TRIP(ISO))
      DO 213 IS=1,50
      IF (IS.GT.KPLT) GO TO 213
      ALGTC(IS) = (GPLT(ISO)-PISO(ISO))/(1.0+VALUM(ISO)*PISO(ISO))
      IF (ALGTC(IS).LT.1.0.OR.ALGTC(IS).GT.5.0) GO TO 213
      KP = KP+1
      ALSPE(KP) = STRL(ISO)
      ALTPE(KP) = ALGTC(ISO)
C206  WRITE (7,507) VALUA(LA),CISOT(ISO),ALTT(ISO),STRL(ISO),ANLO(ISO)
213  CONTINUE
      IVGR(2) = KP
      IVGR(3) = 2
      IVGR(4) = 0
      CALL GPLOT (ALTPE,ALSPE,IVGR)
211  CONTINUE
C
C  PLOT THE PREDICTED ISOTHERMALS--LOG STRESS VS LOG TIME
C
214  PCUT = .05
      IVGR(2) = 1
      IVGR(3) = 3
      IVGR(4) = 186
      CALL GPLOT (CUTLOG,PCUT,IVGR)
      DO 265 K=1,NT
      KP=0
      TRIP(K) = (((TSTA(K)-TSTA(NP))/(TSTA(NP)+460.))**2
      VALUM(K) = VALUA(LA)*(1.0-AKON*TRIP(K))
      DO 260 I=1,50
      IF (I.GT.KPLT) GO TO 260
      ALGTC(I) = (GPLT(I)-TFCN(K))/(1.0+VALUM(K)*TFCN(K))
      IF (ALGTC(I).LT.1.0.OR.ALGTC(I).GT.5.0) GO TO 260
      KP=KP+1
      ALOGP(KP) = STRL(I)
      ALTP(KP) = ALGTC(I)
C262  WRITE (7,507) VALUA(LA),TSTA(K),ALGTC(I),STRL(I),TFCN(K)
260  CONTINUE
      CALL INTENS(40)
      IVGR(2) = KP
      IVGR(3) = 0
      CALL GPLOT (ALTP,ALOGP,IVGR)
      CALL INTENS(2)
265  CONTINUE
C  CALL ENDID(1,0,GRNAM)
      CALL DISPLA(1)

```

```

C
C   END OF ISOTHERMAL PLOT
C
C
C   PLOT STRESS VS. LOG TIME FOR SELECTED TEMPERATURES
C   SINCE THERE ARE NO PREDICTIONS, USE ONLY A=0
C
256 IF (OPT(2)) GO TO 261
   IF (VALUA(LA).NE.0.0) GO TO 261
   CALL BEGID(2)
   NUMPT = NODTW
   GRAXV(3) = 0.
   GRAXV(4) = 0.5
   GRAXV(5) = 4.5
   GRAXV(6) = .5
   GRAXV(7) = 2.
   GRAXV(8) = -1.
   GRAXV(9) = 0.
   RMIN = 0.0
   RMAX = 12.5
   IF (SSTA(3).GT.1.00) RMAX=50.
   IF (SSTA(3).GT.1.70) RMAX=75.
   IF (SSTA(3).GT.1.88) RMAX=100.
   IF (SSTA(3).GT.2.01) RMAX=125.
   IF (SSTA(3).GT.2.10) RMAX=150.
   IF (SSTA(3).GT.2.18) RMAX=200.
   GRAYV(3) = 90.
   GRAYV(4) = RMIN
   GRAYV(5) = RMAX
   GRAYV(6) = 9.
   CALL XAXIS (.9,.6,GRAXV)
   CALL YAXIS (.9,.6,GRAYV)
   IVGR(3) = 3
   CALL TITLE (1,64,25,TITEL)
   CALL CHARS (16,DATES,0.,1.7,9.4,12)
   CALL TITLE (3,12,20,QLABX8)
   CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
   IA=1
   DO 264 K=1,NISO
     IK=0
     DO 263 I=1,NODTW
       IF (WTMP(I).NE.CISOT(K)) GO TO 263
       IK=IK+1
     237 QXPL(IK)=WALT(I)
       QYPL(IK)=SIG(I)
     263 CONTINUE
C
C   PLOT THE DATA FOR THE K TH TEMPERATURE
C
       IVGR(2) = IK
       IVGR(4) = KSYM(K)
       CALL GPLOT (QXPL,QYPL,IVGR)
     264 CONTINUE
       CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
       CALL TITLE (4,16,20,QLABX1)
       IVGR(2) = 1
       IVGR(4) = 186
       CALL GPLOT (CUTLOG,PCUT,IVGR)
C
       CALL ENDID(2,0,GRNAM)
       CALL DISPLA(1)
C
C   END OF RAW DATA PLOT
C
C   SET UP A TABLE OF LOG G = LOG H VALUES FOR THE CASE OF A=0 ONLY
C
       WRITE (9,481) (XFILS(II),II=1,49,3)
       WRITE (9,481) (YFILS(II),II=1,50,3)

```

```

261 IF (VALUA(LA).NE.0.0) GO TO 251
    DO 266 I=1,50
    DAZERO(I) = GPLAT(I)
    AMATRX(I,LA) = DAZERO(I)
266 CONTINUE
    GO TO 252
C
C SET UP A TABLE OF LOG S AND LOG H VALUES FOR THE OTHER A VALUES
C
251 DO 267 I=1,50
    AMATRX (I,LA) = GPLAT(I)
    BNONZ(I) = GPLAT(I)-ALTT(I)
    BMATRX (I,LA) = BNONZ(I)
267 CONTINUE
C
C PLOT THE G STATION FUNCTION VS LOG STRESS
C
252 SPLPT = SSTA(2)
    SPLY = SFCN(1)
    SPLX = 10.0+*SSTA(2)
    IF (OPT(3)) GO TO 254
    CALL BEGID(3)
    NUMPT = KPLT
    CALL SOLBAK (XAXV,NUMPT,STRL,XMIS)
    CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
    GRAXV(3) = 0.
    GRAXV(4) = XRMIN
    GRAXV(5) = XRMAX
    GRAXV(6) = 10.
    CALL SOLBAK (YAXV,NUMPT,GPLT,YMIS)
    CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
    GRAYV(3) = 90.
    GRAYV(4) = YRMIN
    GRAYV(5) = YRMAX
    GRAYV(6) = 10.
    CALL XAXIS (.9,.6,GRAXV)
    CALL YAXIS (.9,.6,GRAYV)
    IVGR(2) = KPLT
    IVGR(3) = 0
    IVGR(4) = 70
    CALL GPLOT (STRL,GPLT,IVGR)
C WRITE (6,519) ((STRL(I),I=1,6),(GPLT(I),I=1,6))
    IVGR(2) = 1
    IVGR(3) = 3
    IVGR(4) = 71
257 CALL GPLOT (SPLPT,SPLY,IVGR)
    CALL TITLE (1,64,25,TITEL)
    CALL CHARS (16,DATES,0.,1.7,9.4,12)
    CALL TITLE (3,4,20,QLABY2)
    CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
    CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
    CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
    CALL TITLE (4,16,20,QLABY1)
C CALL ENDDID(3,0,GRNAM)
    CALL DISPLA(1)
C
C PLOT THE G STATION FUNCTION VS STRESS
C
254 IF (OPT(4)) GO TO 253
    CALL BEGID(4)
    NUMPT = KPLT
    CALL SOLBAK (XAXV,NUMPT,ALSTRL,XMIS)
    CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
    GRAXV(3) = 0.
    GRAXV(4) = XRMIN
    GRAXV(5) = XRMAX
    GRAXV(6) = 10.
    CALL SOLBAK (YAXV,NUMPT,GPLT,YMIS)
    CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)

```

```

GRAYV(3) = 90.
GRAYV(4) = YRMIN
GRAYV(5) = YRMAX
GRAYV(6) = 10.
CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = KPLT
IVGR(3) = 0
IVGR(4) = 70
255 CALL GPLOT (ALSTRL,GPLT,IVGR)
C WRITE (6,519) ((ALSTRL(I),I=1,6),(GPLT(I),I=1,6))
IVGR(2) = 1
IVGR(3) = 3
IVGR(4) = 71
CALL GPLOT (SPLX,SPLY,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,4,20,QLABY2)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITL (4,12,20,QLABX8)
C CALL ENDID(4,0,GRNAM)
CALL DISPLA(1)
C
C PLOT THE P STATION FUNCTION VS TEMP
C
253 TFIL(1) = TSTA(1)
DTFIL(1) = TSTA(1)
PFIL(1) = TFCN(1)
TRFIL(1) = 1000./(TSTA(1)+460.)
TDEL = ( TSTA(NT) - TSTA(1) )/ 30.0
249 NFIL = 31
DO 269 I=2,NFIL
TFIL(I) = TFIL(I-1) + TDEL
DTFIL(I) = TFIL(I)
TRFIL(I) = 1000./(TFIL(I)+460.)
248 PFIL(I) = D+E*TFIL(I)+F*TFIL(I)**2
269 CONTINUE
C WRITE (7,487) (TFIL(I),I=1,NFIL)
C WRITE (7,491) (PFIL(I),I=1,NFIL)
258 IF (OPT(5)) GO TO 259
CALL BEGID(5)
NUMPT = NFIL
CALL SCLBAK (XAXV,NUMPT,TFIL,XMIS)
CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
GRAXV(3) = 0.
GRAXV(4) = XRMIN
GRAXV(5) = XRMAX
GRAXV(6) = 10.
CALL SCLBAK (YAXV,NUMPT,PFIL,YMIS)
CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
GRAYV(3) = 90.
GRAYV(4) = YRMIN
GRAYV(5) = YRMAX
GRAYV(6) = 10.
CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = NFIL
IVGR(3) = 0
IVGR(4) = 70
CALL GPLOT (TFIL,PFIL,IVGR)
IVGR(2) = NT
IVGR(3) = 3
IVGR(4) = 70
CALL GPLOT (QXPLTT,QYPLTT,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,4,20,QLABY3)

```

```

      CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
      CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
      CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
      CALL TITLE (4,4,20,QLABX3)
C     CALL ENDID(5,0,GRNAM)
      CALL DISPLA(1)
C
C PLOT THE P STATION FUNCTION VS RECIPROCAL TEMP
C
259 IF (OPT(6)) GO TO 268
      CALL BEGID(6)
      NUMPT = NFIL
      CALL SCLBAK (XAXV,NUMPT,TRFIL,XMIS)
      CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
      GRAXV(3) = 0.
      GRAXV(4) = XRMIN
      GRAXV(5) = XRMAX
      GRAXV(6) = 10.
      CALL SCLBAK (YAXV,NUMPT,PFIL,YMIS)
      CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
      GRAYV(3) = 90.
      GRAYV(4) = YRMIN
      GRAYV(5) = YRMAX
      GRAYV(6) = 10.
      CALL XAXIS (.9,.6,GRAXV)
      CALL YAXIS (.9,.6,GRAYV)
      IVGR(2) = NFIL
      IVGR(3) = 0
      IVGR(4) = 70
      CALL GPLOT (TRFIL,PFIL,IVGR)
      IVGR(2) = NT
      IVGR(3) = 3
      IVGR(4) = 70
      CALL GPLOT (QXPLRT,QYPLTT,IVGR)
      CALL TITLE (1,64,25,TITEL)
      CALL CHARS (16,DATES,0.,1.7,9.4,12)
      CALL TITLE (3,4,20,QLABY3)
      CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
      CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
      CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
      CALL TITLE (4,28,20,QLABX4)
C     CALL ENDID(6,0,GRNAM)
      CALL DISPLA(1)
C
C PLOT THE MASTER CURVE LOG SIGMA VS. THE PARAMETER G
C
C NOTA BENE MEG16B
C
C IF TEMPERATURE IS NOT A STATION VALUE, USE QUADRATIC CURVE FIT
C TO INTERPOLATE TEMP FUNCTION IN CALCULATION OF PARAMETER
C
268 DO 285 I=1,NODT
      DO 270 IT=1,NT
        IY=IT
        IF (WTMP(I).EQ.TSTA(IT)) GO TO 275
270 CONTINUE
        PEST(I)=D+E*WTMP(I)+F*WTMP(I)**2
        GO TO 280
275 PEST(I)=TFCN(IY)
        TRIP(I)=((WTMP(I)-TSTA(NP))/(TSTA(NP)+460.))**2
        VALUM(I)=VALUA(LA)*(1.0-AKON*TRIP(I))
280 GCALC(I)=WALT(I)+PEST(I)*(VALUM(I)*WALT(I)+1.0)
285 CONTINUE
286 IF (OPT(7)) GO TO 274
      CALL BEGID(7)
      NUMPT = KPLT
      CALL SCLBAK (XAXV,NUMPT,GPLT,XMIS)
      CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
      GRAXV(3) = 0.

```



```

GRAXV(4) = XRMIN
GRAXV(5) = XRMAX
GRAXV(6) = 10.
CALL SOLBAK (YAXV,NUMPT,STRL,YMIS)
CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
GRAYV(3) = 90.
GRAYV(4) = YRMIN
GRAYV(5) = YRMAX
GRAYV(6) = 10.
CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = 1
IVGR(3) = 3
IVGR(4) = 156
SPLY = .6
CALL GPLOT (SPLY,SPLPT,IVGR)
IVGR(2) = KPLT
IVGR(3) = 0
CALL GPLOT (GFLT,STRL,IVGR)
IVGR(2) = NODT
IVGR(3) = 3
IVGR(4) = 62
CALL GPLOT (GCALC,QALS,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,16,20,QLABY1)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8.2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,28,20,QLABX5)
CALL NUMBER (2,SFCN(4),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,2.0,15)
CALL NUMBER (2,SFCN(3),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,2.4,15)
CALL NUMBER (2,SFCN(2),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,2.8,15)
CALL NUMBER (2,SFCN(1),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,3.2,15)
CALL NUMBER (2,QYPLTT(NT),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,3.6,15)
CALL NUMBER (2,QYPLTT(2),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,4.0,15)
CALL NUMBER (2,QYPLTT(1),12,3,GRLEG1)
CALL CHARS (12,GRLEG1,0.,1.1,4.4,15)
C CALL ENDDID(7,0,GRNAM)
CALL DISPLA(1)
C
274 IF (OPT(8)) GO TO 281
C
281 IF (OPT(9)) GO TO 277
C
C CALCULATE STANDARD DEVIATION OF DATA USED--IN TERMS OF LOG TIME (STDEV)
C
277 CONTINUE
SUMSQ=0.0
SUMRES=0.0
SUMAR=0.0
SUMSQR = 0.0
SUMZD = 0.0
SUMZD2 = 0.0
IR = 1
WRITE (9,459) VALUA(LA)
WRITE (9,475)
DO 325 J=1,NODUW
YVAL = WALS(J)
288 CALL TUSTR (SSTA,YVAL,NQ,ALPHL,ALPHH,COEFGG)
C WRITE (7,450) (COEFGG(KK),KK=1,NS)
YVALU = WTMP(J)
321 CALL STACH (TSTA,YVALU,NT,COEFP,N)

```

```

C      WRITE (7,450) (COEFP(KK),KK=1,NT)
      PTRM=COEFP(N-1)*TFCN(N-1)+COEFP(N)*TFCN(N)+COEFP(N+1)*TFCN(N+1)+COEFP-
325 1(N+2)*TFCN(N+2)
      TRIP(J)=((WTMP(J)-TSTA(NP))/(TSTA(NP)+460.))*2
      VALUM(J)=VALUA(LA)*(1.0-AKON*TRIP(J))
      HOURS(J)=(GPLAT(J)-PTRM)/(1.0+VALUM(J)*PTRM)
      RES(J)=WALT(J)-HOURS(J)
      SUMSQ=SUMSQ+RES(J)*RES(J)
      SUMRES=SUMRES+RES(J)
      ABRES(J)=DABS(WALT(J)-HOURS(J))
      SUMAR=SUMAR+ABRES(J)
      ZDEV(J) = WALT(J)*(1.0+VALUM(J)*PTRM)+PTRM-GPLAT(J)
      SUMZD = SUMZD + ZDEV(J)
      SUMZD2 = SUMZD2 + ZDEV(J)*ZDEV(J)
      WRITE (9,489) WALT(J),HOURS(J),RES(J),SUMSQ,WTMP(J),WALS(J),GPLAT(J),PTRM,N
325  CONTINUE
C
C      NOTA BENE      MEG16B
C
C      NO. OF PARAMETERS FITTED      = 4+NT-1      SINCE TFCN AT ONE TEMP=0
C      NO. OF DEGREES OF FREEDOM      = NO. OF DATA - (4+NT-1) - 1
C
      NPAR = NT+NQ-1
      NDIV=NODUW-NPAR-1
      IF (NDIV.LT.1) NDIV=1
      DOF=NDIV
      STDEV=SQRT(SUMSQ/DOF)
      SDZD=SQRT(SUMZD2/DOF)
      PSDZD(LA) = SDZD*2.0
      TOT=NODUW
      AVDEV=SUMAR/TOT
      PSTDEV(LA)=STDEV
C326  WRITE (6,485) STDEV,DOF,SUMRES,AVDEV
C
C      CALCULATE RESIDUALS AND NORMAL DEVIATES      FOR ALL DATA      (RSDEV)
C
      WRITE (9,483) TITEL
      WRITE (9,457) VALUA(LA)
      WRITE (9,485) STDEV,DOF,SUMRES,AVDEV
      SMSQL=0.0
      SMARSL=0.0
      DO 375 J=1,NODTW
      YVALU = WTMP(J)
351  CALL STACH (TSTA,YVALU,NT,COEFP,N)
      PTRM=COEFP(N-1)*TFCN(N-1)+COEFP(N)*TFCN(N)+COEFP(N+1)*TFCN(N+1)+COEFP-
1(N+2)*TFCN(N+2)
      TRIP(J)=((WTMP(J)-TSTA(NP))/(TSTA(NP)+460.))*2
      VALUM(J)=VALUA(LA)*(1.0-AKON*TRIP(J))
      HOUR(J)=(GPLAT(J)-PTRM)/(1.0+VALUM(J)*PTRM)
      RESD(J)=WALT(J)-HOUR(J)
      IF (LA.NE.1) GO TO 354
      PLRESD(J) = RESD(J)
354  RSDEV(J)=RESD(J)/STDEV
370  CONTINUE
375  CONTINUE
      WRITE (9,495) WTMP(J),WALS(J),WALT(J),HOUR(J),RESD(J),RSDEV(J)
C      WRITE (6,500) STD1,DENL
C      WRITE (6,505) AVRESL,AVARSL
C
C      PLOT OBSERVED VERSUS PREDICTED VALUES OF LOG TIME TO FAILURE
C
379  IF (OPT(10)) GO TO 385
      CALL BEGID(10)
      GRAXV(4) = 0.5
      GRAXV(5) = 5.5
      GRAXV(6) = 10.
      GRAYV(4) = .5
      GRAYV(5) = 5.5
      GRAYV(6) = 10.

```

```

CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = NODUW
IVGR(3) = 3
IVGR(4) = 63
CALL GPLOT (HOURS,QALT,IVGR)
IVGR(2) = 2
IVGR(3) = 0
CALL GPLOT (QPX,QPY,IVGR)
CALL GPLOT (QPXL,QPYL,IVGR)
CALL GPLOT (QPXR,QPYR,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,4,20,QLABY7)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,4,20,QLABX7)
380 CALL CHARS (12,QSDKEY,0.,7.,1.4,15)
CALL NUMBER (4,STDEV,8,2,GRLEG1)
CALL CHARS (8,GRLEG1,0.,8.4,1.4,15)
C CALL ENDID(10,0,GRNAM)
CALL DISPLA(1)
C
C
385 CONTINUE
C
C END OF THE LOOP FOR VARIOUS A VALUES
C
C THE NEXT FEW PLOTS WILL EVALUATE THE VARIOUS A VALUES USED
C
C
381 IF (OPT(11)) GO TO 382
C
382 IF (OPT(12)) GO TO 383
C
383 IF (OPT(13)) GO TO 384
C
C PLOT THE A SCAN -- S. D. VERSUS A VALUES CHOSEN
C
384 IF (OPT(14)) GO TO 376
DO 396 I=1,LAK
QVALUA(I) = VALUK(I)
396 CONTINUE
WRITE (7,511) QVALUA
C WRITE (7,509) PSTDEV,PSTD
CALL BEGID(14)
GRAXV(4) = -.2
GRAXV(5) = .2
GRAXV(6) = 8.
GRAYV(4) = 0.0
GRAYV(5) = .8
GRAYV(6) = 8.
CALL XAXIS (.9,.6,GRAXV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = LAK
IVGR(3) = 3
IVGR(4) = 66
CALL GPLOT (QVALUA,PSTDEV,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,4,20,QLABYA)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8,2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,8,20,QLABXA)
C CALL ENDID(14,0,GRNAM)
CALL DISPLA(1)

```

```

C
C   WRITE (7,523) (PSDZD(J),J=1,LAK)
C   WRITE (7,502) (PSTDEV(J),J=1,LAK)
C
C   PLOT THE RESIDUALS VERSUS TEMPERATURE
C
376 IF (OPT(15)) GO TO 377
C
  CALL BEGID(15)
  NUMPT = NODTW
  CALL SCLBAK (XAXV,NUMPT,QTMP,XMIS)
  CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
  GRAXV(1) = 10.
  GRAXV(2) = -1.
  GRAXV(3) = 0.
  GRAXV(4) = XRMIN
  GRAXV(5) = XRMAX
  GRAXV(6) = .5
  GRAXV(7) = 2.
  GRAXV(8) = -1.
  GRAXV(9) = 0.
  CALL SCLBAK (YAXV,NUMPT,PLRES,YMIS)
  CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
  GRAYV(1) = 10.
  GRAYV(2) = -1.
  GRAYV(3) = 90.
  GRAYV(4) = -1.0
  GRAYV(5) = 1.0
  GRAYV(6) = .5
  GRAYV(7) = 2.
  GRAYV(8) = -1.
  GRAYV(9) = 0.
  CALL XAXIS (.9,.6,GRAXV)
  CALL YAXIS (.9,.6,GRAYV)
  IVGR(1) = 7
  IVGR(2) = NUMPT
  IVGR(3) = 3
  IVGR(4) = 62
  IVGR(5) = 1
  IVGR(6) = 15
  IVGR(7) = 0
  CALL GPLOT (QTMP,PLRES,IVGR)
  CALL TITLE (1,64,25,TITEL)
  CALL CHARS (16,DATES,0.,1.7,9.4,12)
  CALL TITLE (3,20,20,QLABY4)
  CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
  CALL NUMBER (4,VALUA(LA),8.2,GRLEG6)
  CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
  CALL TITLE (4,4,20,QLABX3)
  CALL ENDID(15,0,GRNAM)
  CALL DISPLA(1)
C
C   PLOT THE RESIDUALS VERSUS LOG STRESS
C
377 IF (OPT(16)) GO TO 378
C
  CALL BEGID(16)
  NUMPT = NODTW
  CALL SCLBAK (XAXV,NUMPT,QALS,XMIS)
  CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
  GRAXV(3) = 0.
  GRAXV(4) = XRMIN
  GRAXV(5) = XRMAX
  GRAXV(6) = .5
  CALL SCLBAK (YAXV,NUMPT,PLRES,YMIS)
  CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
  GRAYV(3) = 90.
  GRAYV(4) = -1.0
  GRAYV(5) = 1.0

```

```

GRAYV(5) = .5
CALL XAXIS (.9,.6,GRAYV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = NUMPT
IVGR(3) = 3
IVGR(4) = 62
CALL GPLOT (QALS,PLRES,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,20,20,QLABY4)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8.2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,12,20,QLABX2)
C CALL ENDID(16,0,GRNAM)
CALL DISPLA(1)
C
C PLOT THE RESIDUALS VERSUS LOG LIFE
C
378 IF (OPT(17)) GO TO 394
C
CALL BEGID(17)
NUMPT = NUDIW
CALL SOLBAK (XAXV,NUMPT,QALT,XMIS)
CALL GINTVL (XMIS(1),XMIS(2),10,0,XRMIN,XRMAX)
GRAYV(3) = 0.
GRAYV(4) = XRMIN
GRAYV(5) = XRMAX
GRAYV(6) = .5
CALL SOLBAK (YAXV,NUMPT,PLRES,YMIS)
CALL GINTVL (YMIS(1),YMIS(2),10,1,YRMIN,YRMAX)
GRAYV(3) = 90.
GRAYV(4) = -1.0
GRAYV(5) = 1.0
GRAYV(6) = .5
CALL XAXIS (.9,.6,GRAYV)
CALL YAXIS (.9,.6,GRAYV)
IVGR(2) = NUMPT
IVGR(3) = 3
IVGR(4) = 62
CALL GPLOT (QALT,PLRES,IVGR)
CALL TITLE (1,64,25,TITEL)
CALL CHARS (16,DATES,0.,1.7,9.4,12)
CALL TITLE (3,20,20,QLABY4)
CALL CHARS (8,GRLEG5,0.,8.0,9.5,15)
CALL NUMBER (4,VALUA(LA),8.2,GRLEG6)
CALL CHARS (8,GRLEG6,0.,8.5,9.5,15)
CALL TITLE (4,16,20,QLABX1)
C CALL ENDID(17,0,GRNAM)
CALL DISPLA(1)
C
C 394 IF (NXAV.NE.1) GO TO 395
LAK = LAK-NXAV
C CALL TERM
395 GO TO 100
C
C
C
C INPUT FORMATS
C
400 FORMAT (16F5.0)
15 IF (K-6) 30,25,20
20 ANS=Z*Z
ANS=ANS**3
GO TO 65
25 ANS=Z*Z
ANS=Z*ANS**2
GO TO 65

```

```

30 IF (K-4) 45,40,35
35 ANS=Z*Z
   ANS=ANS*ANS
   GO TO 65
40 ANS=Z*Z*Z
   GO TO 65
45 IF (K-2) 60,55,50
50 ANS=Z*Z
   GO TO 65
55 ANS=Z
   GO TO 65
60 ANS=1.
65 FUNC=ANS
   RETURN

C
70 FORMAT (1H0,5X,72HTOO MANY COEFFICIENTS ASKED FOR - ONLY 8 ARE A-
1AVAILABLE YOU CANNOT GET I3)
   END

C.....
402 FORMAT (16I5)
404 FORMAT (2I5,F9.1,F6.3,4X,17L1)
408 FORMAT (F5.0,F7.3,F8.2,F5.0)
410 FORMAT (16A4)

C
C OUTPUT FORMATS
C
455 FORMAT (1H1,50X,18A4)
457 FORMAT (1H0,50X,22HTHE VALUE OF 'A' = ,F7.3)
459 FORMAT (1H0,5X,25H DATA USED IN CALCULATION,22X,22HTHE VALUE OF 'A' -
1 = ,F7.3)
461 FORMAT (1H0,5X,27H LONG TIME DATA NOT USED ,20X,22HTHE VALUE OF 'A' -
1 = ,F7.3)
463 FORMAT (1H0,1X,10HG STATIONS/10X,10F10.3/)
465 FORMAT (1H0,1X,24H G COEFFICIENTS /10X,10F10.3/)
469 FORMAT (1H0,1X,24H P STATION FUNCTION /10X,10F10.3/)
467 FORMAT (1H0,1X,10HP STATIONS/10X,10F10.3)
471 FORMAT (1H0,5X,3HNO.,5X,11HTEMPERATURE,9X,6HSTRESS,13X,4HTIME,9X,1-
11HLOG(STRESS),9X,9HLOG(TIME)/)
475 FORMAT (1H0,20X,2HLOG TIME,22X,6H TEMP,2X,10HLOG STRESS , -
1 3X,3H G ,7X,3H P /11X,4H OBS,6X,5H PRED,5X,5H DIFF,5X,6HSUM SQ )
477 FORMAT (1H0,50HTHE NF,NP,NG,MK, NL,NT,NS,NISO,NXAV VALUES ARE ,4I-
15,5X,5I5)
481 FORMAT (1X,18F7.3)
483 FORMAT (1H1,20X,18A4)
485 FORMAT (1H0,16HS. D. OF RES = ,F6.3,2X,6HDOF = ,F6.1,2X,14H-
1SUM OF RES = ,F6.3,2X,14HAM ABS DEV = ,F6.3)
487 FORMAT (1H0,7HTEMP = /10X,10F10.3)
489 FORMAT (8X,4F10.5,F10.1,3F10.5,14)
491 FORMAT (1H0,7HP(T) = /10X,10F10.3)
495 FORMAT (6X,F10.0,3F12.5,F12.4,F12.3)
500 FORMAT (1H0,35HSTANDARD ERROR OF PREDICTED VALUES = ,F7.3,3X,26H-
1NO. OF DATA PREDICTED = ,F6.1)
501 FORMAT (1H0,11F10.4)
502 FORMAT (1X,'S. D. OF DATA USED = ',9F8.3)
503 FORMAT (1X,'S. D. OF DATA PRED. = ',9F8.3)
504 FORMAT (1H0,1X,'CUTOFF = ',2F11.2,3X,'REDUCED DATA BEGIN AT LIFE = ', -
1 F11.2,3X,F6.2,2X,F5.0)
505 FORMAT (1X,39HAGG. DEVIATION OF PREDICTED VALUES = ,F7.3,3X,39H-
1AGG. ABS. DEV. OF PREDICTED VALUES = ,F7.3)
506 FORMAT (1H0,1X,'NO. OF DATA USED = ',16,5X,'NO. WEIGHTED = ', -
1 16,5X,'NO. PREDICTED = ',16,5X,'D O F = ',F6.0 )
507 FORMAT (4X,F5.2,F10.0,3X,F8.3,3X,F10.4,2X,F10.4)
509 FORMAT (5X,9F6.3,2X,9F5.3)
511 FORMAT (2X,23HTHE VALUES OF A = ,9F8.3)
513 FORMAT (3X,54HA VALUE TEMP LOG TIME LOG S STRESS )
515 FORMAT (1H1,2X,3CHRESULTS FROM MEG16B (MAY 78) / -
1 15X,39HLOGH + A LOGH * P(T) + P(T) = G(SIGMA) )
516 FORMAT (10X,4A4)
517 FORMAT (2X,12F6.2)

```

```

518 FORMAT (2X,12I6)
519 FORMAT (2X,12F8.3)
523 FORMAT (1X,'S. D. OF ZERO DEV.   = ',9F8.3)
524 FORMAT (1X,11F9.3)
C
399 STOP
C
END
C .....
C
SUBROUTINE TISTR (XS,X,NS,ALPL,ALPH,COEF)
C
C THIS PROGRAM SETS UP THE STRESS FUNCTION IN TERMS OF
C TWO STRAIGHT LINES MERGED TOGETHER SMOOTHLY
C
C XS IS THE VECTOR OF 'STATION VALUES' ONLY XS(2) IS USED IN THIS VERSION
C X IS THE VALUE OF LOG STRESS TO BE EXPRESSED IN TERMS OF THE MODEL EQ.
C NS IS THE NUMBER OF STATION VALUES SET = 3 USUALLY
C ALPL IS THE EXPONENT FOR THE LOW STRESS REGION
C ALPH IS THE EXPONENT FOR THE HIGH STRESS REGION
C COEF IS THE VECTOR OF COEFFICIENTS FOR EACH X VALUE
C
DOUBLE PRECISION XS,X,COEF,S,SLOG,SSQ,C1,C2,C3
DIMENSION XS(4),COEF(4)
WRITE (7,200) XS
10 S = 10.0*(X-XS(2))
SLOG = DLOG10(10.0*X/10.0*XS(2))
SSQ = S**2
C1 = 2.302585093
C2 = 1.0/(C1*ALPL)
C3 = 1.0/(C1*ALPH)
IF (X.GT.XS(2)) GO TO 20
C
C FOR STRESS LT MID STRESS REGION 1
C
15 COEF(1) = 1.0
COEF(2) = SLOG
COEF(3) = C2*(C2*S**ALPL-C2-SLOG)
COEF(4) = 0.0
GO TO 30
C
C FOR STRESS GT MID STRESS REGION 2
C
20 COEF(1) = 1.0
COEF(2) = SLOG
COEF(3) = 0.0
COEF(4) = C3*(C3*S**ALPH-C3-SLOG)
30 CONTINUE
C WRITE (7,204) X
C WRITE (7,201) COEF
C WRITE (7,203) S,SLOG,SSQ
200 FORMAT (1X,3F8.4)
201 FORMAT (10X,5F10.5)
203 FORMAT (1X,4F9.5)
204 FORMAT (10X,' X = ',F7.4)
RETURN
END
C .....
C
SUBROUTINE STACH (X,Y,NS,COEF,N)
C
C TO EXPRESS A SINGLE VALUE OF A VARIABLE IN TERMS OF DISCRETE 'STATIONS'
C
C X IS THE VECTOR OF 'STATION VALUES'
C Y IS THE VALUE TO BE EXPRESSED IN TERMS OF STATION VALUES
C NS IS THE NUMBER OF STATION VALUES
C COEF IS THE VECTOR OF COEFFICIENTS FOR EACH Y VALUE
C

```



```

C GIVEN A TABLE OF DISCRETE X VALUES AND ONE VALUE, CALLED Y ,
C FIND, FOR THE Y VALUE, THE COEFFICIENTS OF A TYPE OF LAGRANGIAN
C INTERPOLATION EQUATION
C
  DOUBLE PRECISION X,Y,COEF,CA,CB,CC,CD,CE,CF,CH,CJ,CK
  DIMENSION COEF(32),X(32)
C WRITE (7,205) X
C WRITE (7,205) Y
  5  CA = 0.0
    CB = 0.0
    CC = 0.0
    CD = 0.0
    CE = 0.0
    CF = 0.0
    CH = 0.0
    CJ = 0.0
    CK = 0.0
    DO 10 K=1,32
  10  COEF(K) = 0.0
  30  DO 20 K= 1,NS
    M=NS-K
C   WRITE (7,206) K,M
    IF (Y.LT.X(2)) GO TO 15
    IF (Y.GE.X(NS-1)) GO TO 16
    IF (Y.GE.X(M)) GO TO 17
  20  CONTINUE
  17  N=M
    CA = Y-X(N-1)
    CB = Y-X(N)
    CC = Y-X(N+1)
    CD = Y-X(N+2)
    CF = X(N )-X(N+1)
    CE = X(N )-X(N-1)
    CH = X(N+1)-X(N-1)
    CJ = X(N )-X(N+2)
    CK = X(N+1)-X(N+2)
  40  COEF(N-1) = (CB*CC)/(CE*CH)*0.5
    COEF(N+2) = (CB*CC)/(CJ*CK)*0.5
    COEF(N ) = ((CA*CC*CJ)+(CC*CD*CE))/(CE*CF*CJ)*0.5
    COEF(N+1) = -((CA*CB*CK)+(CB*CD*CH))/(CF*CH*CK)*0.5
    GO TO 45
  15  N=2
    GO TO 18
  16  N=NS-1
  18  COEF(N-1) = (X(N)-Y)*(X(N+1)-Y)/((X(N)-X(N-1))*(X(N+1)-X(N-1)))
    COEF(N) = (X(N+1)-Y)*(Y-X(N-1))/((X(N+1)-X(N))*(X(N)-X(N-1)))
    COEF(N+1) = (X(N)-Y)*(Y-X(N-1))/((X(N)-X(N+1))*(X(N+1)-X(N-1)))
  45  CONTINUE
C   WRITE (7,206) N
C   WRITE (6,205) Y
C   WRITE (6,205) (COEF(L),L=1,NS)
  205  FORMAT (1X, 10F10.4)
  206  FORMAT (1X, 3I4)
    RETURN
    END
C.....

```

```

C      SUBROUTINE AMATF (M,N,MK,KSPARS,U)
C
C      MLSS/MFSS PROGRAM BEING USED TO SOLVE THE REDUNDANT EQS
C
C      THE U MATRIX MUST BE BROKEN UP INTO THE PROPER MATRICES
C
C      M      IS THE NUMBER OF COLUMNS IN MATRIX
C      N      IS THE NUMBER OF ROWS
C      MK      IS THE COLUMN TO BE USED AS DEPENDENT VARIABLE
C      KS      IS A COUNTER NOT USED IN THIS VERSION
C      U      IS THE MATRIX TO BE INVERTED
C      XF      IS THE SOLUTION VECTOR
C
C      DOUBLE PRECISION XF
C      DOUBLE PRECISION F(200),AUT(2304),TRAC(200),U(200,48),A(48,200),XXT(48,48)
C      COMMON /CCOMN/ XF(48),NP
C      DO 5 I=1,48
C      DO 5 J=1,200
C      A(I,J)=0.
C
C      READ X
C
C      MKM1=MK-1
C      MKP1=MK+1
C      DO 20 I=1,N
C      DO 10 J=1,MKM1
C      A(J,I)=U(I,J)
C      DO 15 J=MKP1,M
C      A(J-1,I)=U(I,J)
C      WRITE (6,60) (U(I,J),J=1,M)
C      F(I)=-U(I,MK)
C      CONTINUE
C      M=M-1
C
C      WRITE (6,60)((A(J,I),J=1,M),I=1,N)
C
C      T
C      FORM XX
C
C      DO 25 I=1,M
C      DO 25 J=1,M
C      XXT(I,J)=0.
C      DO 25 K=1,N
C      XXT(I,J)=XXT(I,J)+A(I,K)*A(J,K)
C      DO 7 I=1,M
C      7 WRITE (6,60) (XXT(I,J),J=1,M)
C
C      FORM XF
C
C      DO 30 I=1,M
C      XF(I)=0.
C      DO 30 K=1,N
C      XF(I)=XF(I)+A(I,K)*F(K)
C
C      PREPARE FOR LEAST SQUARES
C
C      EPS=1.E-5
C      K=0
C      DO 35 J=1,M
C      DO 35 I=1,J
C      K=K+1
C      35 AUT(K)=XXT(I,J)
C      KTOT=K
C      WRITE (6,45) (AUT(K),K=1,KTOT)
C      36 CALL DMFSS (AUT,M,EPS,IRANK,TRAC)
C      WRITE (6,50) IRANK
C      WRITE (6,45) (AUT(K),K=1,KTOT)

```

```

C      COMPUTE      T -1
C      A= (XX ) XF, THE SOLUTION OF XF - XX A= 0
C
C 37 CALL DMLSS (AUT,M,IRANK,TRAC,0,XF,IER)
C      WRITE (6,50) IER
C      WRITE (6,45) (XF(I),I=1,M)
C
C      RETURN
C
C 40 FORMAT (1H1/5H M= , I3,3X,4HN= , I3/1X/2H X/1X)
C 45 FORMAT (2H F/1X/(8E16.8))
C 50 FORMAT (15)
C 60 FORMAT (2X,10F8.3)
C      END
C .....
C
C      SUBROUTINE DMFSS
C
C      PURPOSE
C      GIVEN A SYMMETRIC POSITIVE SEMI DEFINITE MATRIX ,DMFSS WILL
C      (1) DETERMINE THE RANK AND LINEARLY INDEPENDENT ROWS AND
C          COLUMNS
C      (2) FACTOR A SYMMETRIC SUBMATRIX OF MAXIMAL RANK
C      (3) EXPRESS NONBASIC ROWS IN TERMS OF BASIC ONES,
C          EXPRESS NONBASIC COLUMNS IN TERMS OF BASIC ONES
C          EXPRESS BASIC VARIABLES IN TERMS OF FREE ONES
C      SUBROUTINE DMFSS MAY BE USED AS A PREPARATORY STEP FOR THE
C      CALCULATION OF THE LEAST SQUARES SOLUTION OF MINIMAL
C      LENGTH OF A SYSTEM OF LINEAR EQUATIONS WITH SYMMETRIC
C      POSITIVE SEMI-DEFINITE COEFFICIENT MATRIX
C
C      USAGE
C      CALL DMFSS(A,N,EPS,IRANK,TRAC)
C
C      DESCRIPTION OF PARAMETERS
C      A      - UPPER TRIANGULAR PART OF GIVEN SYMMETRIC SEMI-
C              DEFINITE MATRIX STORED COLUMNWISE IN COMPRESSED FORM
C              ON RETURN A CONTAINS THE MATRIX T AND, IF IRANK IS
C              LESS THAN N, THE MATRICES U AND TU
C              A MUST BE OF DOUBLE PRECISION
C      N      - DIMENSION OF GIVEN MATRIX A
C      EPS    - TESTVALUE FOR ZERO AFFECTED BY ROUND-OFF NOISE
C      IRANK  - RESULTANT VARIABLE, CONTAINING THE RANK OF GIVEN
C              MATRIX A IF A IS SEMI-DEFINITE
C              IRANK = 0 MEANS A HAS NO POSITIVE DIAGONAL ELEMENT
C              AND/OR EPS IS NOT ABSOLUTELY LESS THAN ONE
C              IRANK = -1 MEANS DIMENSION N IS NOT POSITIVE
C              IRANK = -2 MEANS COMPLETE FAILURE, POSSIBLY DUE TO
C              INADEQUATE RELATIVE TOLERANCE EPS
C      TRAC   - VECTOR OF DIMENSION N CONTAINING THE
C              SOURCE INDEX OF THE I-TH PIVOT ROW IN ITS I-TH
C              LOCATION, THIS MEANS THAT TRAC CONTAINS THE
C              PRODUCT REPRESENTATION OF THE PERMUTATION WHICH
C              IS APPLIED TO ROWS AND COLUMNS OF A IN TERMS OF
C              TRANSPOSITIONS
C              TRAC MUST BE OF DOUBLE PRECISION
C
C      REMARKS
C      EPS MUST BE ABSOLUTELY LESS THAN ONE. A SENSIBLE VALUE IS
C      SOMEWHERE IN BETWEEN 10**(-4) AND 10**(-6)
C      THE ABSOLUTE VALUE OF INPUT PARAMETER EPS IS USED AS
C      RELATIVE TOLERANCE.
C      IN ORDER TO PRESERVE SYMMETRY ONLY PIVOTING ALONG THE
C      DIAGONAL IS BUILT IN.
C      ALL PIVOTELEMENTS MUST BE GREATER THAN THE ABSOLUTE VALUE
C      OF EPS TIMES ORIGINAL DIAGONAL ELEMENT
C      OTHERWISE THEY ARE TREATED AS IF THEY WERE ZERO
C      MATRIX A REMAINS UNCHANGED IF THE RESULTANT VALUE IRANK
C      EQUALS ZERO

```

```

C
C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C      NONE
C
C      METHOD
C      THE SQUARE ROOT METHOD WITH DIAGONAL PIVOTING IS USED FOR
C      CALCULATION OF THE RIGHT HAND TRIANGULAR FACTOR.
C      IN CASE OF AN ONLY SEMI-DEFINITE MATRIX THE SUBROUTINE
C      RETURNS THE IRANK X IRANK UPPER TRIANGULAR FACTOR T OF A
C      SUBMATRIX OF MAXIMAL RANK, THE IRANK X (N-IRANK) MATRIX U
C      AND THE (N-IRANK) X (N-IRANK) UPPER TRIANGULAR TU SUCH
C      THAT TRANSPOSE(TU)*TU=I+TRANSPOSE(U)*U
C
C      .....
C
C      SUBROUTINE DMFSS(A,N,EPS,IRANK,TRAC)
C
C      DIMENSIONED DUMMY VARIABLES
C      DIMENSION A(1),TRAC(1)
C      DOUBLE PRECISION SUM,A,TRAC,PIV,HOLD
C
C      TEST OF SPECIFIED DIMENSION
C      IF(N)36,36,1
C
C      INITIALIZE TRIANGULAR FACTORIZATION
C      1 IRANK=0
C      ISUB=0
C      KPIV=0
C      J=0
C      PIV=0.D0
C
C      SEARCH FIRST PIVOT ELEMENT
C      DO 3 K=1,N
C      J=J+K
C      TRAC(K)=A(J)
C      IF(A(J)-PIV)3,3,2
C      2 PIV=A(J)
C      KSUB=J
C      KPIV=K
C      3 CONTINUE
C
C      START LOOP OVER ALL ROWS OF A
C      DO 32 I=1,N
C      ISUB=ISUB+I
C      IM1=I-1
C      4 KMI=KPIV-I
C      IF(KMI)35,9,5
C
C      PERFORM PARTIAL COLUMN INTERCHANGE
C      5 JI=KSUB-KMI
C      IDC=JI-ISUB
C      JJ=ISUB-IM1
C      DO 6 K=JJ,ISUB
C      KK=K+IDC
C      HOLD=A(K)
C      A(K)=A(KK)
C      6 A(KK)=HOLD
C
C      PERFORM PARTIAL ROW INTERCHANGE
C      KK=KSUB
C      DO 7 K=KPIV,N
C      II=KK-KMI
C      HOLD=A(KK)
C      A(KK)=A(II)
C      A(II)=HOLD
C      7 KK=KK+K

```

```

C
C      PERFORM REMAINING INTERCHANGE
      JJ=KPIV-1
      II=ISUB
      DO 8 K=I,JJ
      HOLD=A(II)
      A(II)=A(JI)
      A(JI)=HOLD
      II=II+K
8     JI=JI+1
9     IF(IRANK)22,10,10
C
C      RECORD INTERCHANGE IN TRANSPOSITION VECTOR
10    TRAC(KPIV)=TRAC(I)
      TRAC(I)=KPIV
C
C      MODIFY CURRENT PIVOT ROW
      KK=IM1-IRANK
      KMI=ISUB-KK
      PIV=0.D0
      IDC=IRANK+1
      JJ=ISUB-1
      JK=KMI
      JJ=ISUB-I
      DO 19 K=I,N
      SUM=0.D0
C
C      BUILD UP SCALAR PRODUCT IF NECESSARY
      IF(KK)13,13,11
11    DO 12 J=KMI,JI
      SUM=SUM-A(J)*A(JK)
12    JK=JK+1
13    JJ=JJ+K
      IF(K-I)14,14,16
14    SUM=A(ISUB)+SUM
C
C      TEST RADICAND FOR LOSS OF SIGNIFICANCE
      IF(SUM-DABS(A(ISUB)*DBLE(EPS)))20,20,15
15    A(ISUB)=DSQRT(SUM)
      KPIV=I+1
      GOTO 19
16    SUM=(A(JK)+SUM)/A(ISUB)
      A(JK)=SUM
C
C      SEARCH FOR NEXT PIVOT ROW
      IF(A(JJ))19,19,17
17    TRAC(K)=TRAC(K)-SUM*SUM
      HOLD=TRAC(K)/A(JJ)
      IF(PIV-HOLD)18,19,19
18    PIV=HOLD
      KPIV=K
      KSUB=JJ
19    JK=JJ+IDC
      GOTO 32
C
C      CALCULATE MATRIX OF DEPENDENCIES U
20    IF(IRANK)21,21,37
21    IRANK=-1
      GOTO 4
22    IRANK=IM1
      II=ISUB-IRANK
      JI=II
      DO 26 K=1,IRANK
      JI=JI-1
      JK=ISUB-1
      JJ=K-1
      DO 26 J=I,N
      IDC=IRANK
      SUM=0.D0

```

```

      KMI=JI
      KK=JK
      IF(JJ)25,25,23
23  DO 24 L=1,JJ
      IDC=IDC-1
      SUM=SUM-A(KMI)*A(KK)
      KMI=KMI-IDC
24  KK=KK-1
25  A(KK)=(SUM+A(KK))/A(KMI)
26  JK=JK+J
C
C      CALCULATE I+TRANPOSE(U)*U
      JJ=ISUB-I
      PIV=0.D0
      KK=ISUB-1
      DO 31 K=I,N
      JJ=JJ+K
      IDC=0
      DO 28 J=K,N
      SUM=0.D0
      KMI=JJ+IDC
      DO 27 L=II,KK
      JK=L+IDC
27  SUM=SUM+A(L)*A(JK)
      A(KMI)=SUM
28  IDC=IDC+J
      A(JJ)=A(JJ)+1.D0
      TRAC(K)=A(JJ)
C
C      SEARCH NEXT DIAGONAL ELEMENT
      IF(PIV-A(JJ))29,30,30
29  KPIV=K
      KSUB=JJ
      PIV=A(JJ)
30  II=II+K
      KK=KK+K
31  CONTINUE
      GOTO 4
32  CONTINUE
33  IF(IRANK)35,34,35
34  IRANK=N
35  RETURN
C
C      ERROR RETURNS
C
C      RETURN IN CASE OF ILLEGAL DIMENSION
36  IRANK=-1
      RETURN
C
C      INSTABLE FACTORIZATION OF I+TRANPOSE(U)*U
37  IRANK=-2
      RETURN
      END
C.....
C
C      SUBROUTINE DMLSS
C
C      PURPOSE
C      SUBROUTINE DMLSS IS THE SECOND STEP IN THE PROCEDURE FOR
C      CALCULATING THE LEAST SQUARES SOLUTION OF MINIMAL LENGTH
C      OF A SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS WITH SYMMETRIC
C      POSITIVE SEMI-DEFINITE COEFFICIENT MATRIX.
C
C      USAGE
C      CALL DMLSS(A,N,IRANK,TRAC,INC,RHS,IER)
C

```

DESCRIPTION OF PARAMETERS

A - COEFFICIENT MATRIX IN FACTORED FORM AS GENERATED BY SUBROUTINE MFSS FROM INITIALLY GIVEN SYMMETRIC COEFFICIENT MATRIX A STORED IN $N*(N+1)/2$ LOCATIONS A REMAINS UNCHANGED A MUST BE OF DOUBLE PRECISION

N - DIMENSION OF COEFFICIENT MATRIX

IRANK - RANK OF COEFFICIENT MATRIX, CALCULATED BY MEANS OF SUBROUTINE DMFSS

TRAC - VECTOR OF DIMENSION N CONTAINING THE SUBSCRIPTS OF PIVOT ROWS AND COLUMNS, I.E. THE PRODUCT REPRESENTATION IN TRANSPOSITIONS OF THE PERMUTATION WHICH WAS APPLIED TO ROWS AND COLUMNS OF A IN THE FACTORIZATION PROCESS TRAC IS A RESULTANT ARRAY OF SUBROUTINE MFSS TRAC MUST BE OF DOUBLE PRECISION

INC - INPUT VARIABLE WHICH SHOULD CONTAIN THE VALUE ZERO IF THE SYSTEM OF SIMULTANEOUS EQUATIONS IS KNOWN TO BE COMPATIBLE AND A NONZERO VALUE OTHERWISE

RHS - VECTOR OF DIMENSION N CONTAINING THE RIGHT HAND SIDE ON RETURN RHS CONTAINS THE MINIMAL LENGTH SOLUTION RHS MUST BE OF DOUBLE PRECISION

IER - RESULTANT ERROR PARAMETER
IER = 0 MEANS NO ERRORS
IER = -1 MEANS N AND/OR IRANK IS NOT POSITIVE AND/OR IRANK IS GREATER THAN N
IER = 1 MEANS THE FACTORIZATION CONTAINED IN A HAS ZERO DIVISORS AND/OR TRAC CONTAINS VALUES OUTSIDE THE FEASIBLE RANGE 1 UP TO N

REMARKS

THE MINIMAL LENGTH SOLUTION IS PRODUCED IN THE STORAGE LOCATIONS OCCUPIED BY THE RIGHT HAND SIDE. SUBROUTINE DMLSS DOES TAKE CARE OF THE PERMUTATION WHICH WAS APPLIED TO ROWS AND COLUMNS OF A. OPERATION IS BYPASSED IN CASE OF A NON POSITIVE VALUE OF IRANK

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

NONE

METHOD

LET T, U, TU BE THE COMPONENTS OF THE FACTORIZATION OF A, AND LET THE RIGHT HAND SIDE BE PARTITIONED INTO A FIRST PART X1 OF DIMENSION IRANK AND A SECOND PART X2 OF DIMENSION $N-IRANK$. THEN THE FOLLOWING OPERATIONS ARE APPLIED IN SEQUENCE

- (1) INTERCHANGE RIGHT HAND SIDE
- (2) $X1 = X1 + U * X2$
- (3) $X2 = -TRANSPOSE(U) * X1$
- (4) $X2 = INVERSE(TU) * INVERSE(TRANSPOSE(TU)) * X2$
- (5) $X1 = X1 + U * X2$
- (6) $X1 = INVERSE(T) * INVERSE(TRANSPOSE(T)) * X1$
- (7) $X2 = -TRANSPOSE(U) * X1$
- (8) $X2 = INVERSE(TU) * INVERSE(TRANSPOSE(TU)) * X2$
- (9) $X1 = X1 + U * X2$
- (10) $X2 = TRANSPOSE(U) * X1$
- (11) REINTERCHANGE CALCULATED SOLUTION

IF THE SYSTEM OF SIMULTANEOUS LINEAR EQUATIONS IS SPECIFIED TO BE COMPATIBLE THEN STEPS (2), (3), (4) AND (5) ARE CANCELLED.

IF THE COEFFICIENT MATRIX HAS RANK N, THEN THE ONLY STEPS PERFORMED ARE (1), (6) AND (11).

```

C      SUBROUTINE DMLSS(A,N,IRANK,TRAC,INC,RHS,IER)
C
C      DIMENSIONED DUMMY VARIABLES
C      DIMENSION A(1),TRAC(1),RHS(1)
C      DOUBLE PRECISION SUM,A,RHS,TRAC,HOLD
C
C      TEST OF SPECIFIED DIMENSIONS
C      IDEF=N-IRANK
C      IF(N)33,33,1
C      1 IF(IRANK)33,33,2
C      2 IF(IDEF)33,3,3
C
C      CALCULATE AUXILIARY ALUES
C      3 ITE=IRANK*(IRANK+1)/2
C      IX2=IRANK+1
C      NP1=N+1
C      IER=0
C
C      INTERCHANGE RIGHT HAND SIDE
C      JJ=1
C      II=1
C      4 DO 6 I=1,N
C      J=TRAC(II)
C      IF(J)31,31,5
C      5 HOLD=RHS(II)
C      RHS(II)=RHS(J)
C      RHS(J)=HOLD
C      6 II=II+JJ
C      IF(JJ)32,7,7
C
C      PERFORM STEP 2 IF NECESSARY
C      7 ISW=1
C      IF(INC*IDEF)8,28,8
C
C      CALCULATE  $X1 = X1 + U * X2$ 
C      8 ISTA=ITE
C      DO 10 I=1,IRANK
C      ISTA=ISTA+1
C      JJ=ISTA
C      SUM=0.D0
C      DO 9 J=IX2,N
C      SUM=SUM+A(JJ)*RHS(J)
C      9 JJ=JJ+J
C      10 RHS(I)=RHS(I)+SUM
C      GOTO(11,28,11),ISW
C
C      CALCULATE  $X2 = TRANSPOSE(U) * X1$ 
C      11 ISTA=ITE
C      DO 15 I=IX2,N
C      JJ=ISTA
C      SUM=0.D0
C      DO 12 J=1,IRANK
C      JJ=JJ+1
C      12 SUM=SUM+A(JJ)*RHS(J)
C      GOTO(13,13,14),ISW
C      13 SUM=-SUM
C      14 RHS(I)=SUM
C      15 ISTA=ISTA+I
C      GOTO(16,29,30),ISW
C
C      INITIALIZE STEP (4) OR STEP (8)
C      16 ISTA=IX2
C      IEND=N
C      JJ=ITE+ISTA
C
C      DIVISION OF  $X1$  BY TRANSPOSE OF TRIANGULAR MATRIX
C      17 SUM=0.D0
C      DO 20 I=ISTA,IEND

```



```

      IF(A(JJ))18,31,18
18  RHS(I)=(RHS(I)-SUM)/A(JJ)
      IF(I-IEND)19,21,21
19  JJ=JJ+1
      SUM=0.D0
      DO 20 J=ISTA,I
      SUM=SUM+A(JJ)*RHS(J)
20  JJ=JJ+1
C
C      DIVISION OF X1 BY TRIANGULAR MATRIX
21  SUM=0.D0
      II=IEND
      DO 24 I=ISTA,IEND
      RHS(II)=(RHS(II)-SUM)/A(JJ)
      IF(II-ISTA)25,25,22
22  KK=JJ-1
      SUM=0.D0
      DO 23 J=II,IEND
      SUM=SUM+A(KK)*RHS(J)
23  KK=KK+J
      JJ=JJ-II
      II=II-1
25  IF(IDEF)26,30,26
26  GOTO(27,11,8),ISW
C
C      PERFORM STEP (5)
27  ISW=2
      GOTO 8
C
C      PERFORM STEP (6)
28  ISTA=1
      IEND=IRANK
      JJ=1
      ISW=2
      GOTO 17
C
C      PERFORM STEP (8)
29  ISW=3
      GOTO 16
C
C      REINTERCHANGE CALCULATED SOLUTION
30  II=N
      JJ=-1
      GOTO 4
C
C      ERROR RETURN IN CASE OF ZERO DIVISOR
31  IER=1
32  RETURN
C
C      ERROR RETURN IN CASE OF ILLEGAL DIMENSION
33  IER=-1
      RETURN
      END
C
C .....
C      SUBROUTINE TOPLT (IP,NT,ALOGS,NQ,GS)
C
C      THIS PROGRAM SPLITS THE XF VECTOR INTO TEMPERATURE AND
C      STRESS COMPONENTS FOR PLOTTING AND CURVE FITTING
C
      DOUBLE PRECISION XF,IP,ALOGS,GS,X,Y,XX,YY,P
      DOUBLE PRECISION XPLT1,TECN,XPLTRI,XPLTIG,YPLTIG
      DIMENSION IP(32),ALOGS(16),P(7),X(32),Y(48),CISOT(32),XX(16),YY(48),GS(16)
      COMMON /ACOMN/ QXPLT1(32),QXPLTIG(32),QYPLT1(32),QYPLTIG(16),QYPLTIG(16)
      COMMON /BCOMN/ XPLT1(32),TECN(32),XPLTRI(32),XPLTIG(16),YPLTIG(16)
      COMMON /CCOMN/ XF(48),NP
      COMMON /DCOMN/ D,E,F,G,H,AA

```



```

C      P = OUTPUT ARRAY OF COEFFICIENTS WHERE P(1) IS THE CONSTANT
C      TERM, P(2) IS THE COEFFICIENT OF THE FIRST DEGREE TERM AND
C      P(3) IS THE COEFFICIENT OF THE SECOND DEGREE TERM, ETC.
C      XP = INPUT ARRAY INDEPENDENT VARIABLE
C      YP = INPUT ARRAY DEPENDENT VARIABLE
C      NBIG = DEGREE OF POLYNOMIAL TO BE ESTIMATED
C      NSMALL = NUMBER OF OBSERVATIONS (NO. PAIRS OF DATA PROVIDED)
C      IPL = CVFIT OUTPUT CONTROL
C              0 FOR NO OUTPUT
C              1 FOR DATA OUTPUT
C              2 FOR DATA AND PLOT OUTPUT
C
C      DETNUM= VARIATION DUE TO REGRESSION
C      DETDEN= TOTAL VARIATION
C      DEV= VARIANCE
C      DVIN= STD. DEVIATION
C      DETRM= COEFFICIENT OF DETERMINATION
C      CORRL= CORRELATION COEFFICIENT
C      ERRATA= RELATIVE ERROR
C      DELY= RESIDUAL ERROR
C      ERS= RELATIVE VARIANCE
C      ERA= RELATIVE STD. DEVIATION
C
C      DATA PK(1),KPL(1),KPL(2)/1.0,64,2/
C      DOUBLE PRECISION A,D,CC,XMAXX,XP,YP,X,P,Y
1      IF (NSMALL.GT.100) GO TO 65
      KN=NBIG
2      IF (KN.GT.6) GO TO 75
      NBIG = NBIG+1
4      SN=NSMALL
      D(NBIG+1)=0.
3      DO 5 I=1,NSMALL
      X(I)=XP(I)
      Y(I)=YP(I)
5      D(NBIG+1)=D(NBIG+1)+Y(I)**2
      XMAXX=X(1)
      DO 15 I=2,NSMALL
      IF (XMAXX-DABS(X(I))) 10,15,15
10     XMAXX=DABS(X(I))
15     CONTINUE
C
C      NORMALIZE THE INDEPENDENT VARIABLE TO THE INTERVAL -1.,+1.
C
      DO 20 I=1,NSMALL
20     X(I)=X(I)/XMAXX
C
C      SET UP THE MATRIX OF COEFFICIENTS
C
      DO 25 I=1,NBIG
      DO 25 J=1,NBIG
21     A(I,J)=0.
      DO 25 K=1,NSMALL
22     FI=FUNC(I,X(K))
23     FJ=FUNC(J,X(K))
24     A(I,J)=A(I,J)+FI*FJ
25     CONTINUE
C
      DO 30 I=1,NBIG
      D(I)=0.
      DO 30 K=1,NSMALL
      FI=FUNC(I,X(K))
      D(I)=D(I)+Y(K)*FI
30     CONTINUE
C
      CALL DTRIA (A,NBIG,D,DET)
      CALL DSOLVE (A,NBIG,CC)
      XMAXX=1./XMAXX
      DETNUM=0.
      DO 35 J=1,NBIG

```

```

      P(J)=CC(J)
      DETNUM=DETNUM+CC(J)*D(J)
35    P(J)=P(J)*FUNC(J,XXAXX)
      IF (IPL.EQ.0) RETURN
C
C   COMPUTE Y-CALC, DELTA-Y, AND STANDARD DEVIATION
C
      DETNUM=DETNUM-SUMYSQ
      DETDEN=D(NBIG+1)-SUMYSQ
      DETRM=DETNUM/DETDEN
      CORRL=SQRT(ABS(DETRM))
      CN=NBIG
      ENDIV=SN-CN-1.
      POLREG=DETNUM/ENDIV
      TOTREG=DETDEN/ENDIV
      ERS=0.
      DEV=0.
      DO 45 I=1,NSMALL
      X(I)=XP(I)
      K=NSMALL+I
      Y(K)=0.
      DO 40 J=1,NBIG
      FJ=FUNC(J,X(I))
40    Y(K)=Y(K)+P(J)*FJ
C
      DELY(I)=Y(I)-Y(K)
      ERRATA(I)=DELY(I)/Y(K)
      ERS=ERS+ERRATA(I)**2
45    DEV=DEV+DELY(I)*DELY(I)
      DEV=DEV/ENDIV
      IF (ENDIV.GT.0.0) GO TO 50
      DVTN=1.000
      GO TO 55
50    DVTN=SQRT(DEV)
55    ERS=ERS/SN
      ERA=SQRT(ERS)
C
C   WRITE (7,85) LABX,LABY
C   DO 60 I=1,NSMALL
C   K=NSMALL+I
60    WRITE (7,90) X(I),Y(I),Y(K),DELY(I),ERRATA(I)
C   WRITE (7,95) KN
C 61    WRITE (7,100) (P(I),I=1,NBIG)
C   WRITE (7,105) DEV,DVTN,DETRM,CORRL,ERS,ERA
C   WRITE (7,110) DET
C
      IF (IPL.LT.2) RETURN
C
      RETURN
65    WRITE (6,115) NSMALL
75    WRITE (6,125) NBIG
      STOP
C
85    FORMAT (1X,3A4,12X,2A4,10X,12H CALC FUNC. ,6X,10H DEVIATION,8X,1-
15H RELATIVE ERROR)
90    FORMAT (5G18.8)
95    FORMAT (41H THE REGRESSION EQUATION FOR THE ABOVE IS/43H  $Y = A(0) -$ 
1 + SUM OF ((  $A(J) * X**J$  )), J=1,I1)
100   FORMAT (7G18.8)
105   FORMAT (1X,2X,14H THE VARIANCE=G15.7,20H STANDARD DEVIATION=G15.7-
1/3X,14H DETERMINATION=G15.7,8X,12H CORRELATION=G15.7/1X,2X,14H PCT -
2VARIANCE=G15.7,20H STD. PCT DEVIATION=G15.7)
110   FORMAT (13H DETERMINANT=G14.6//)
115   FORMAT (43H ONLY 100 DATA POINTS ALLOWED, YOUR NSMALL=I13,1H.)
120   FORMAT (6H NBIG=2A4,2X,7H NSMALL=2A4)
125   FORMAT (1X,2X,20H**POLYNOMIAL DEGREE=I5,11H IS TOO BIG)
      END

```

```

C .....
C
SUBROUTINE DTRIA (B,NR,C,DET)
C
C
DIMENSION A(8,8), C(8), B(8,8)
DOUBLE PRECISION A,B,C
KM1=NR
K=NR+1
DO 5 I=1,KM1
DO 5 J=1,KM1
5 A(I,J)=B(I,J)
DO 10 I=1,KM1
10 A(I,K)=C(I)
DO 15 N=2,K
15 A(1,N)=A(1,N)/A(1,1)
DO 40 J=2,K
M=0
L=J-1
DO 40 I=2,KM1
M=M+1
IF (M-L) 25,25,20
20 M=L
25 DO 30 N=1,M
30 A(I,J)=A(I,J)-A(N,J)*A(I,N)
IF (I-J) 35,40,40
35 A(I,J)=A(I,J)/A(I,I)
40 CONTINUE
DET=1.0
DO 45 I=1,KM1
DET=DET*A(I,I)
DO 45 J=1,K
45 B(I,J)=A(I,J)
B(KM1,KM1)=1.0
RETURN
END
C .....
C
SUBROUTINE DSOLVE (A,M,X)
DIMENSION A(8,8), X(7)
DOUBLE PRECISION A,X
MP1=M+1
MM1=M-1
DO 10 K=1,MM1
MMK=M-K
MMKP1=MMK+1
1 X(M)=A(M,MP1)/A(M,M)
SUM=0.
DO 5 I=MMKP1,M
5 SUM=SUM+A(MMK,I)*X(I)
10 X(MMK)=A(MMK,MP1)-SUM
RETURN
END
C .....
C
FUNCTION FUNC (I,X)
DOUBLE PRECISION X,Z,ANS
Z=X
K=I
IF (K-8) 15,10,5
5 WRITE (6,70) K
STOP
10 ANS=ALOG10(Z)
GO TO 65

```

References

1. Goldhoff, R.M., Towards the Standardization of Time-Temperature Parameter Usage in Elevated Temperature Data Analysis. *J. Test. Eval.*, vol. 2, Sep. 1974, pp. 387-424.
2. Manson, S.S.; and Ensign, C.R.: A Quarter-Century of Progress in the Development of Correlation and Extrapolation Methods for Creep Rupture Data. *J. Eng. Mater. Technol.*, vol. 101, no. 4, Oct. 1979, pp. 417-325.
3. Manson, S.S.; and Ensign, C.R.: A Specialized Model for Analysis of Creep-Rupture Data by the Minimum Commitment, Station-Function Approach, NASA TM X-52999, 1971.
4. Manson, S.S.; and Ensign, C.R.: Interpolation and Extrapolation of Creep Rupture Data by the Minimum Commitment Method. Part I—Focal-Point Convergence. Characterization of Materials for Service at Elevated Temperatures, MPC-7, G.V. Smith, ed., American Society of Mechanical Engineers, 1978, pp. 299-398.
5. Graphics System for TSS/360. Manual IB-1310-1002, NASA Lewis Research Center, Computer Systems Office, 1976.

1 Report No. NASA TP-1809	2 Government Accession No.	3 Recipient's Catalog No.	
4 Title and Subtitle MEGA16 - COMPUTER PROGRAM FOR ANALYSIS AND EXTRAPOLATION OF STRESS-RUPTURE DATA		5 Report Date May 1981	
		6 Performing Organization Code 505-33-22	
7 Author(s) C. Robert Ensign		8 Performing Organization Report No. E-495	
		10 Work Unit No.	
9 Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135		11 Contract or Grant No.	
		13 Type of Report and Period Covered Technical Paper	
12 Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14 Sponsoring Agency Code	
15 Supplementary Notes			
16 Abstract <p>The computerized form of the minimum commitment method of interpolating and extrapolating stress versus time to failure data is a program called MEGA16. This report describes MEGA16, giving examples of its many plots and tabular outputs for a typical set of data. The program assumes a specific model equation and then provides a family of predicted isothermals for any set of data with at least 12 stress-rupture results from three different temperatures spread over reasonable stress and time ranges. It is written in Fortran IV using IBM plotting sub-routines and it runs on an IBM 370 time sharing system.</p>			
17 Key Words (Suggested by Author(s)) Stress analysis Creep-rupture strength		18 Distribution Statement Unclassified - unlimited STAR Category 39	
19 Security Classif. (of this report) Unclassified	20 Security Classif. (of this page) Unclassified	21 No. of Pages 46	22 Price* A03

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1981

END

June 15, 1981